National Defense Industrial Association

Files are in Adobe format.
Download the newest version from Adobe.

10th ANNUAL CMMI TECHNOLOGY CONFERENCE AND USER GROUP Tutorial Session

"Investigation, Measures, and Lessons Learned About the Relationship Between CMMI Process Capability and Project or Program Performance"

Denver, CO

15 November 2010

Monday, November 15, 2010

TRACK 1

- 11203 - CMMI® V1.3 and Architecture, Dr. Lawrence Jones, Software Engineering Institute

TRACK 2

- 11288 - Strategic Technology and Operational Risk Management (STORM), Mr. Kobi Vider, K.V.P Consulting

TRACK 3

- 11151 - Making Process Improvement Work - Tying Improvement and CMMI® Directly to What You Care About, Mr. Neil Potter, The Process Group

TRACK 5

- 11262 - SPI Manifesto -Values and Principles, Mr. Tim Kasse, Kasse Initiatives, LLC
- 11263 – Effective Technology Transition Techniques That Make Process Improvement Happen, Mr. Tim Kasse, Kasse Initiatives, LLC

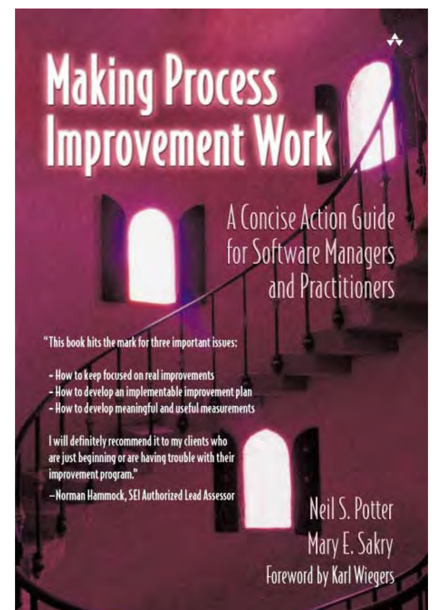
Making Process Improvement Work

Tying Improvement and CMMI® Directly to What You Care About

Neil Potter

Mary Sakry

The Process Group
help@processgroup.com
www.processgroup.com



Agenda - 1

Agenda - 2

Introduction

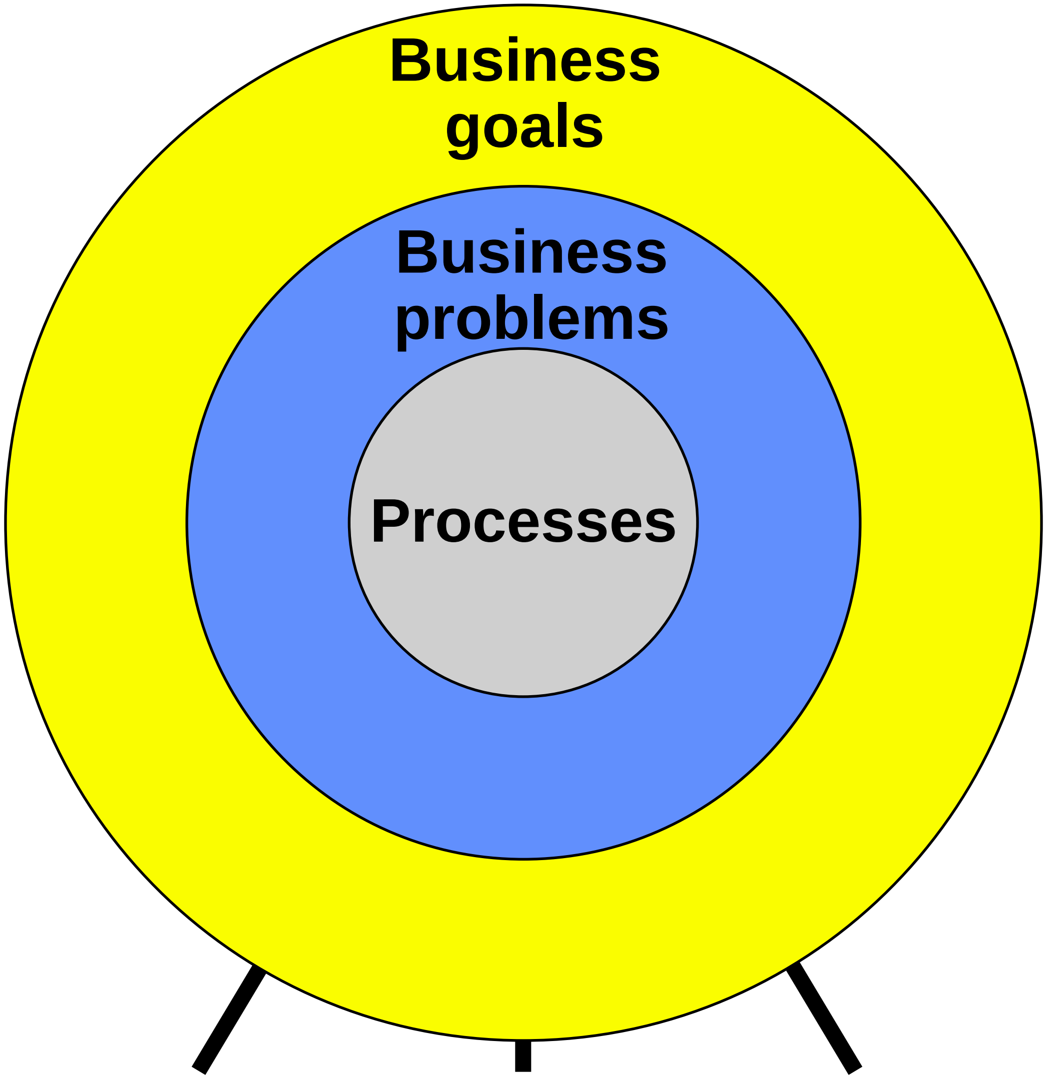
The “Classic” Approach to PI

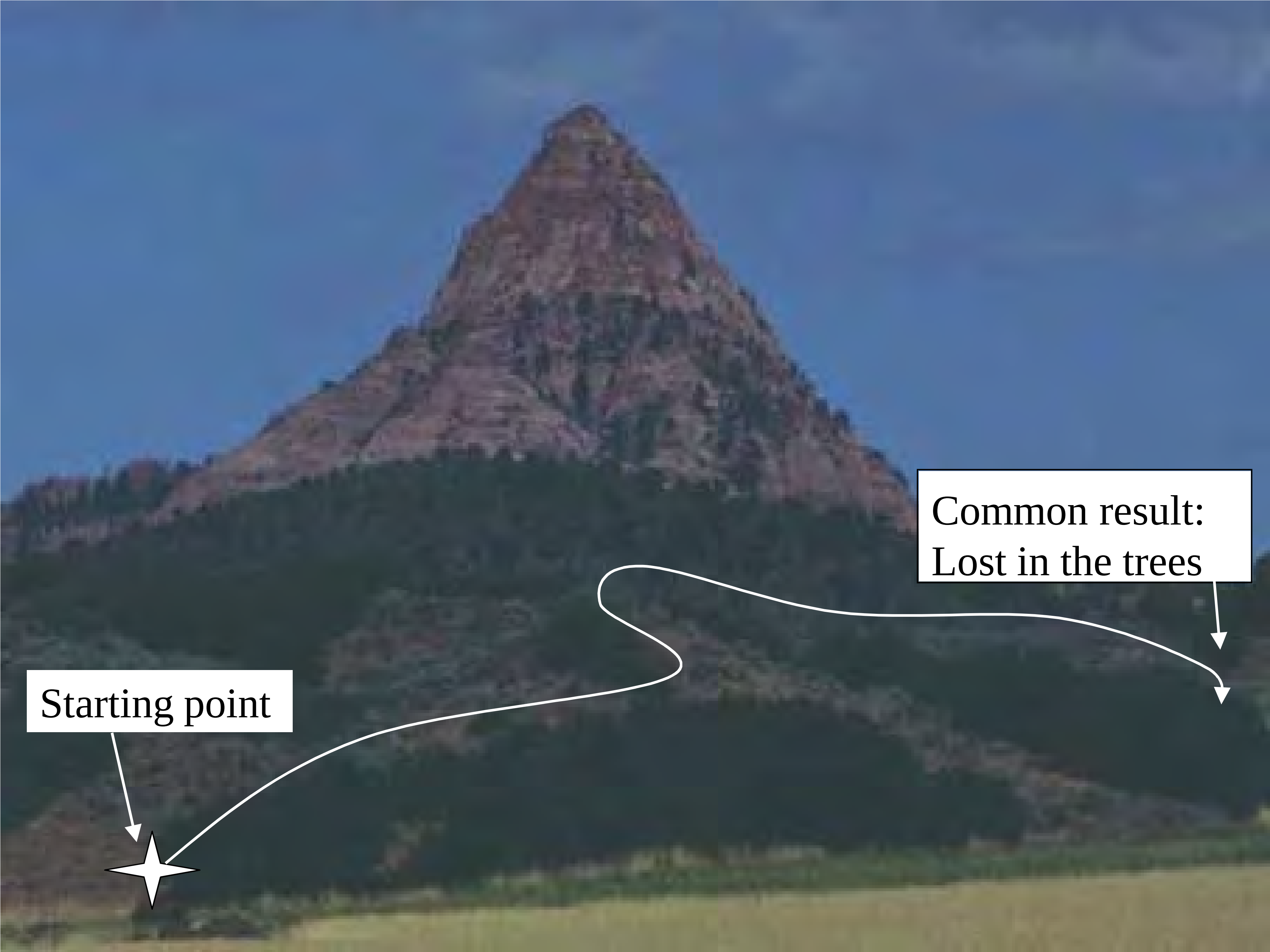
Process-centric improvement

- SEI CMMI
- ISO9001
- Bellcore

It can work!

- High risk of failure

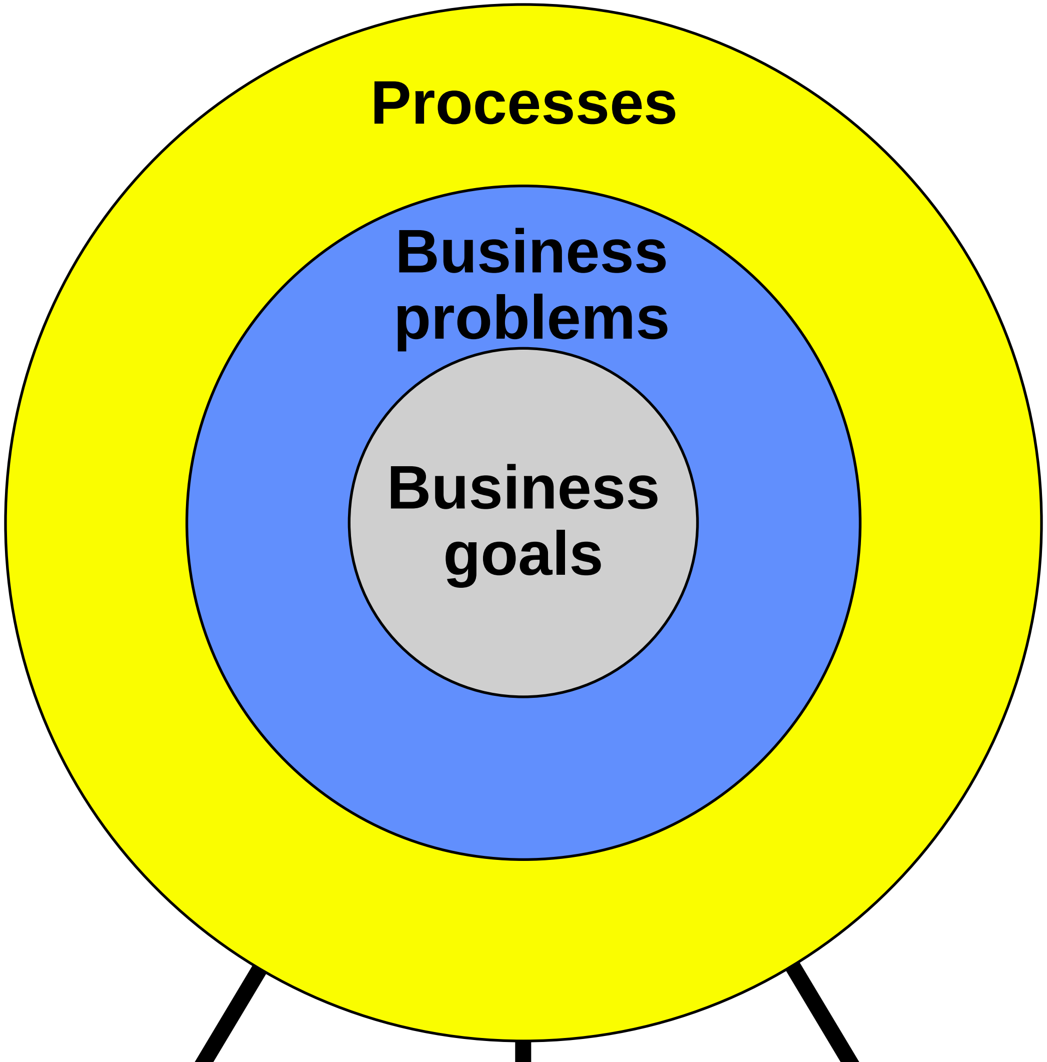



Common result:
Lost in the trees
Starting point

A Solution

Goal-problem-centric improvement

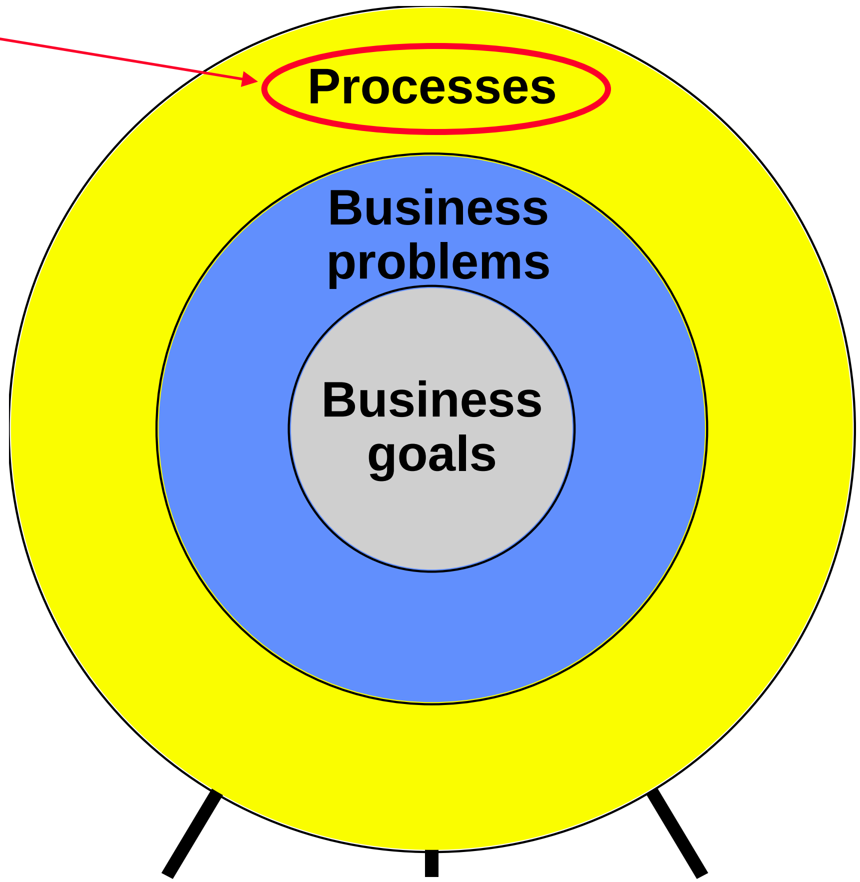
Goals and problems can be used to scope and sequence the improvement effort




Goal
Problem areas
Starting point
• Goal actions
• Improvement actions

Frameworks

- **Frameworks provide an optional source of improvement ideas, e.g.,**
  - Life cycle
  - SEI CMMI
  - ISO9001
  - Bellcore
- **In this workshop, either use:**
  - No framework
  - Current organization's life cycle and defined practices
  - Published framework



Developing a Plan

"Unplanned process improvement is wishful thinking."

—Watts Humphrey, *Managing the Software Process*

Developing a Plan

- **Scope the Improvement**
  1. Establish plan ownership
  2. State the major goals and problems
  3. Group the problems related to each goal
  4. Ensure that the goals and problems are crystal clear and compelling
  5. Set goal priorities
  6. Derive metrics for the goals
- **Develop an Action Plan**
- **Determine Risks and Plan to Mitigate**

THE PRC GRC

1. Establish Plan Ownership

- **The plan meets the owner's needs, e.g.,**
  - Business goals and problems
- **The owner can be a project manager, program manager, senior manager, or division head**
- **The primary owner ≠ EPG or QA group**
  - Support functions can share ownership
- **Different individuals can be responsible for each section of the plan**

EPG = engineering process group
QA = quality assurance group

2. State the Major Goals and Problems - 1

Example Goals

1. Create predictable schedules
2. Successfully deliver product X
3. Reduce rework
4. Improve the performance of our core product
5. Keep customers happy
6. Keep making a profit

State the Major Goals and Problems - 2

Example Problems

1. Need better requirements. Requirements tracking not in place. Changes to requirements are not tracked; code does not match specification at test time.
2. Management direction unclear for product version 2.3. Goals change often.
3. Quality department does not have training in product and test skills.
4. Unclear status of changes.
5. Lack of resources and skills allocated to design.

⋮

9. Defect repairs break essential product features.
10. Wrong files (for example, dynamic link libraries) are put on CD. Unsure of the correct ones.
11. Revising the project plan is difficult. Items drop off, new things are added, plan is out of date.
12. We don't understand our capacity and do not have one list of all the work we have to do.
13. Schedule tracking and communication of changes to affected groups is poor.

3. Group the Problems Related to Each Goal - 1

- **Simplify the list by grouping the problems that prevent each goal from being achieved.**

Goal	Problem	Problem Description
1. Create predictable schedules	Problem 11	Revising the project plan is difficult. Items drop off, new things are added, plan is out of date.
	Problem 12	We don’t understand our capacity and do not have one list of all the work we have to do.
	Problem 13	Schedule tracking and communication of changes to affected groups is poor.

Group the Problems Related to Each Goal - 2

Goal	Problem	Problem Description
2. Successfully deliver product X	Problem 1	Need better requirements. Requirements tracking not in place. Changes to requirements are not tracked; code does not match specification at test time.
	Problem 2	Management direction unclear for product version 2.3. Goals change often.

Ensure That the Goals and Problems Are Compelling - 2

- **Example goals that are not compelling:**
  - Document all processes.
  - Develop a detailed life cycle.
  - Establish a metrics program.
- **Example goals that are more compelling:**
  - Deliver product X by Dec 15th.
  - Increase product quality to a maximum of 10 defects per release, gaining back customers X, Y, and Z, and increasing our market share by 10 percent.
  - Reduce rework to 5 percent of project effort. Use that time to create new product Y.
  - Improve schedule prediction to ± 5-day accuracy, eliminating forced cancellation of vacations.

Ensure That the Goals and Problems Are Crystal Clear

Original Goals	Goals Reworded for Clarity
1. Create predictable schedules	Meet all our cost and schedule commitments
2. Successfully deliver product X	Deliver product X by mm/dd/yy
3. Reduce rework	Reduce rework to less than 20 percent of total project effort
4. Improve the performance of our core product	Improve the performance of our core product (target to be defined)
5. Keep customers happy	Achieve customer rating of 9/10 on product evaluation form
6. Keep making a profit	Keep profits at 15 percent (and costs at the same level as last year)

Using the Approach for a Single Project

What is your goal?

Reduce product development cycle to six to nine months for product X.

What is preventing you from achieving the goal?

1. Changing requirements.
2. Loss of resources; difficult to replace people with specialized skills who leave the project.
3. Too many features for the six- to nine-month development cycle.
4. Poor quality of incoming code from other groups.
5. Inadequate availability of test equipment.
6. Lack of visibility within each life cycle phase. It is difficult to know whether we are ahead or behind schedule.
7. Don’t always have the resources available to complete the planned work.
8. Difficult to find defects early.

Exercise: Scope the Improvement

1. **Form project teams**
2. **Determine the primary business goals and problems of your group**
   – Simplify the list of goals and problems by grouping the related problems under each goal
   – Verify that the scope of your improvement program is compelling
      » If not, ask: Why do I want to achieve these goals?
3. **Discuss lessons learned**

Result:

What is your goal? 3

What is your goal? 2

What is your goal? 1

Reduce product development cycle to six to nine months for product X

What is preventing you from achieving the goal?

1. Changing requirements
2. Loss of resources; difficult to replace people with specialized skills who leave the project
3. Too many features for the six- to nine-month development cycle
4. Poor quality of incoming code from other groups
5. Inadequate availability of test equipment
6. Lack of visibility within each life cycle phase. It is difficult to know whether we are ahead or behind schedule
7. Don Õalways have the resources available to complete the planned work
8. Difficult to find defects early

Developing a Plan

- Scope the Improvement
- <u>Develop an Action Plan</u>
  1. Enumerate actions using brainstorming and a process framework
  2. Organize the action plan based on the goals and problems
  3. Add placeholders for checking progress and taking corrective action
- **Determine Risks and Plan to Mitigate**

Develop an Action Plan

- **Develop an Action Plan**
  1. Enumerate actions using brainstorming and a process framework
     » **1a. What actions are needed to address the problems and achieve the goals?**
     » **1b. If a process improvement framework is being used, which elements will help the problems and goals listed?**
  2. Organize the action plan based on the goals and problems
  3. Add placeholders for checking progress and taking corrective action

1a. Actions for Two of the Problems - 1

Problem	What actions are needed to address the problems and achieve the goals?
1. Changing requirements	Baseline the requirements before design commences
	Only allow changes to the application interface, not to the kernel routines
	Improve the library control system to minimize version control errors
	Investigate requirements management tools

1b. Framework Elements for Two of the Problems - 1

Reworded for clarity

Problem	Which elements will help the problems and goals listed?
1. Changing requirements	Develop an understanding with the requirements providers on the meaning of the requirements. (REQM sp1.1)
	Assign responsibility and authority for performing the REQM process. (REQM gp2.4)
	Track change requests for the configuration items. (CM sp2.1)

REQM = Requirements Management. CM = Configuration Management

Progress on Chosen Framework -1

95% map to Level 2

Example Goals

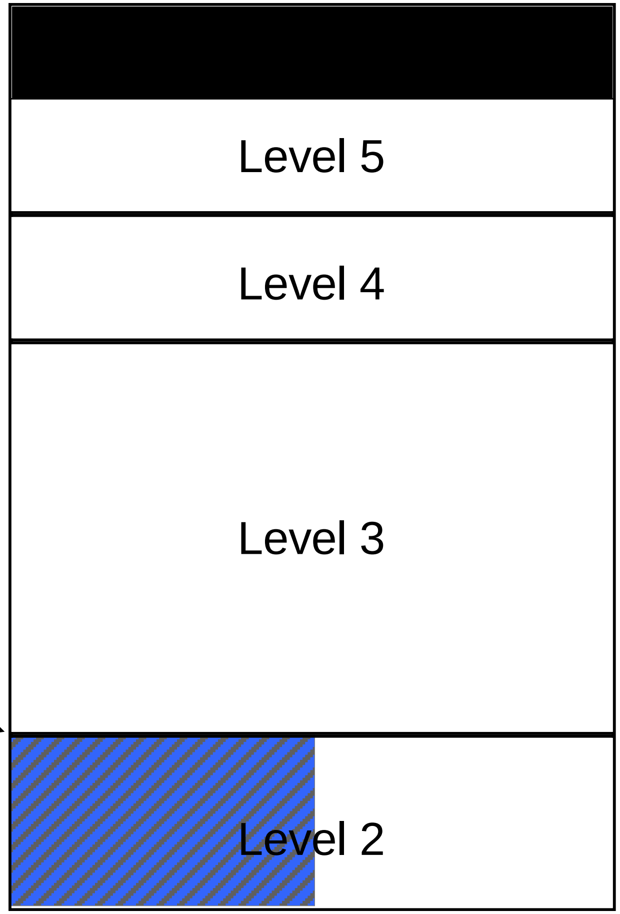
1. Create predictable schedules
2. Successfully deliver product X
3. Reduce rework
4. Improve the performance of our core product
5. Keep customers happy
6. Keep making a profit

Example Problems

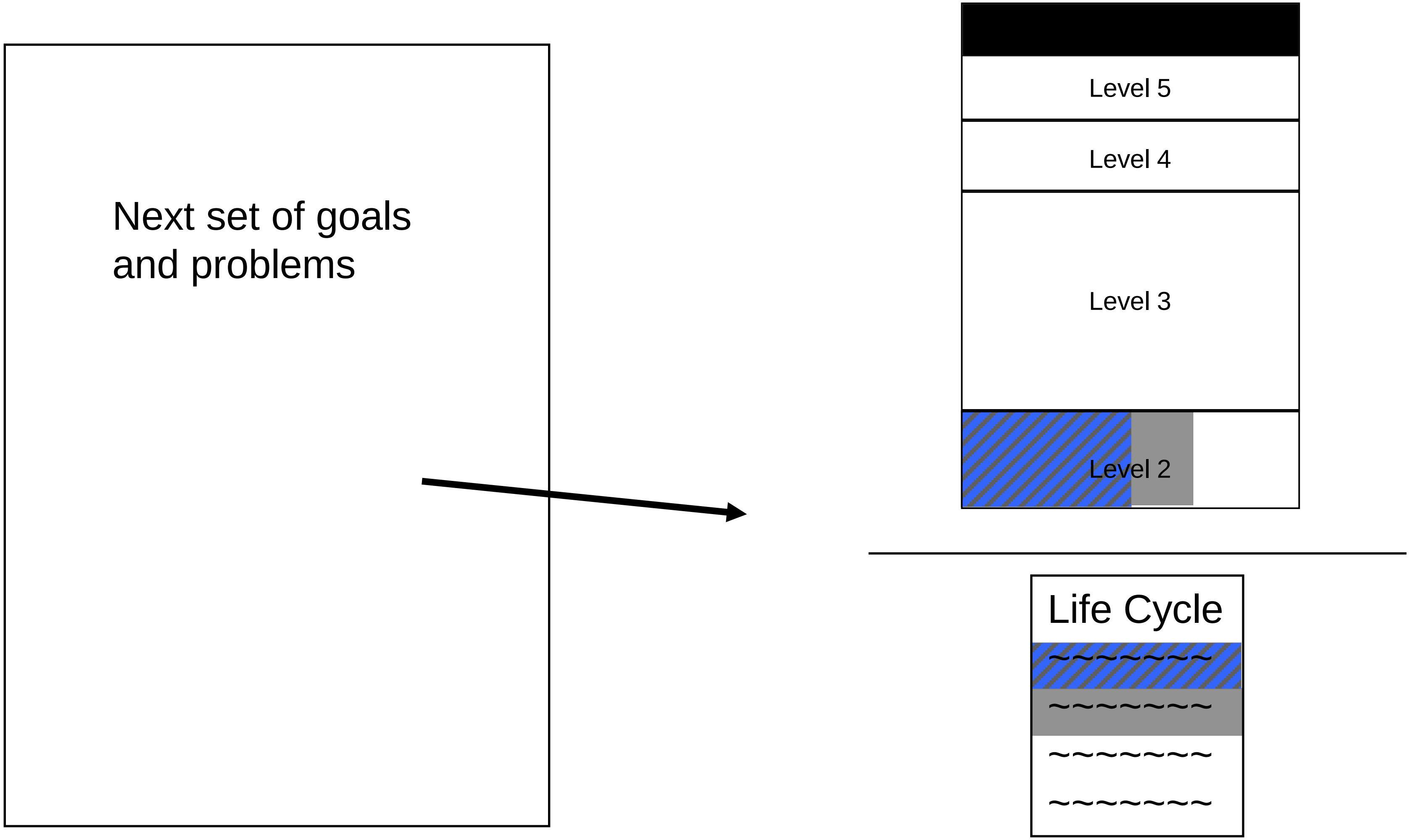
1. Need better requirements. Requirements tracking not in place. Changes to requirements are not tracked; code does not match specification at test time.
2. Management direction unclear for product version 2.3. Goals change often.
3. Quality department does not have training in product and test skills.
4. Unclear status of changes.
5. Lack of resources and skills allocated to design.

9. Defect repairs break essential product features.
10. Wrong files (for example, dynamic link libraries) are put on CD. Unsure of the correct ones.
11. Revising the project plan is difficult. Items drop off, new things are added, plan is out of date.
12. We donÕt understand our capacity and do not have one list of all the work we have to do.
13. Schedule tracking and communication of changes to affected groups is poor.

Initial goals and problems address 43% of Level 2

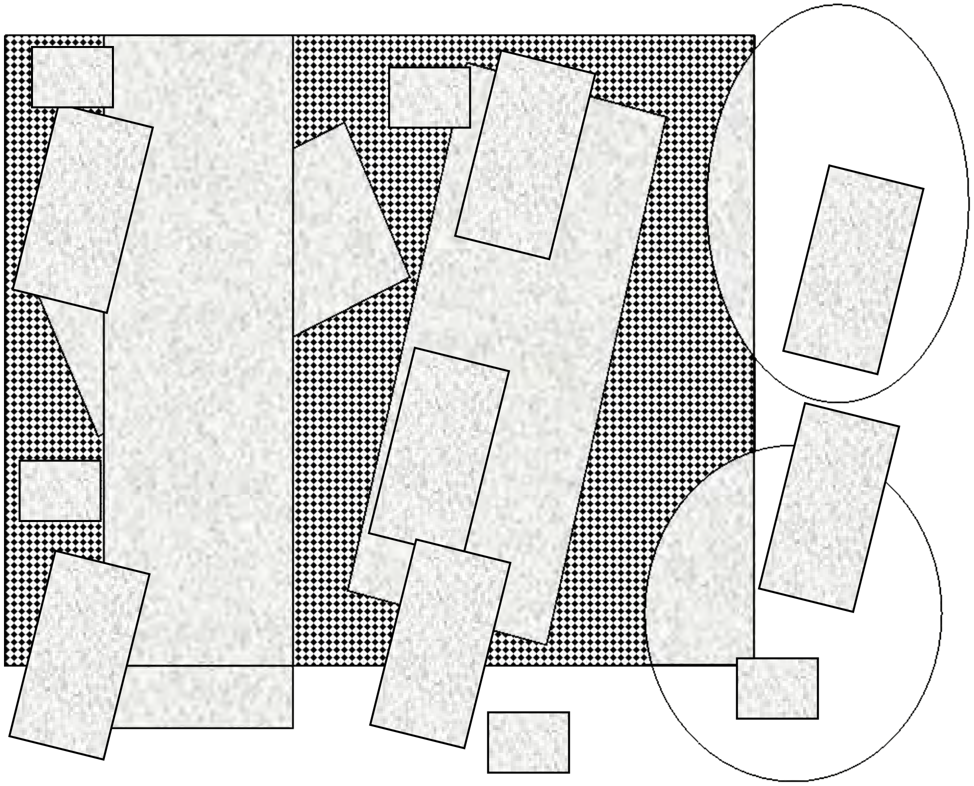


Progress on Chosen Framework -2



What to Do With the Remaining Elements?

- **Put each to good use**
  - –What problem could it solve?
- **Declare them not applicable**
  - –Check with your appraiser / auditor!
- ~~**Meet the letter of the law**~~



2. Organize the Action Plan

Action Plan Owner: ________________

Primary Goal and Intermediate Goals (The result you want)	**Purpose of Goal** (Why do you want to achieve this goal?)	**Actions**	**Priority** (*=essential)	**Time Estimate**	**Who**
PRIMARY GOAL 1	**PURPOSE OF PRIMARY GOAL 1**				
Small intermediate goal (based on problem statement)	Purpose of small intermediate goal	Action	1*		
		Action	2*		
		Action	3		
		Action	4		
Next intermediate goal	Purpose of next intermediate goal	Action	1*		

Template is available at www.processgroup.com/bookinfo.htm.

Example Improvement Plan - 1

Primary Goal and Intermediate Goals (The results you want)	**Purpose of Goal** (Why do you want to achieve the goal?)	**Actions**	**Priority** (*=essential)
Reduce product development cycle to six to nine months for product X.	**Deliver earlier than competition.**		
Manage changing requirements (based on problem 1).	Prevent schedule slips resulting from expensive scope changes.	Only allow changes to the application interface, not the kernel routines.	1*
		Assign responsibility and authority for performing the REQM process.	2*
		Check progress and take corrective action .	-
		Improve the library control system to minimize version control errors. Investigate requirements management tools.	3
		Track change requests for the configuration items.	4
		Develop an understanding with the requirements providers on the meaning of the requirements .	5
		Baseline the requirements before design commences.	6

Step 3: Add placeholder for checking progress and taking corrective action

Summary - Developing a Plan

- **All improvements are tied to specific needs of the organization**
- **Goals and problems help the organization identify which pieces of an improvement framework to implement next**
- **Goals and problems establish the scope and context for each improvement**
  - When a problem has been solved or a goal addressed, a team can stop defining the process or standard
- **Practitioners and managers are motivated to work on improvement because the effort is directed toward the group's needs**

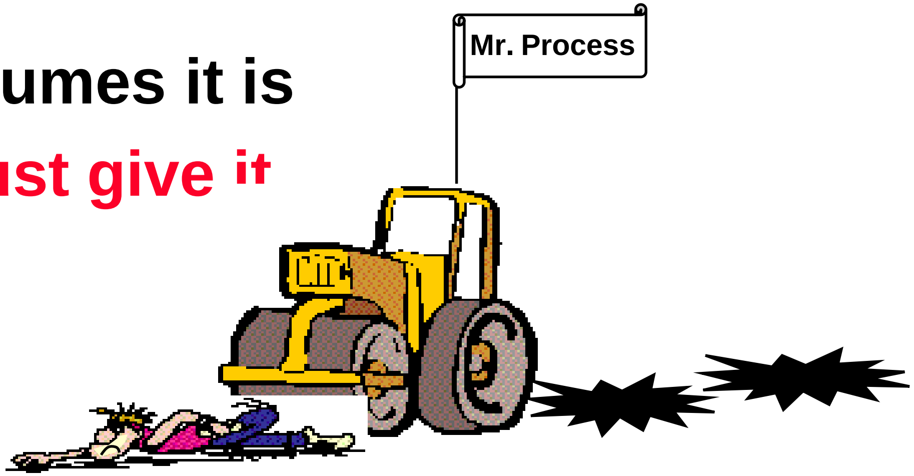
Implementing the Plan

"Proving that the true skeptics are indeed truly skeptical achieves nothing, except that you've dented your pick and probably permanently diminished your credibility (and failed to appreciate the vital importance of building a fragile momentum)."

—Tom Peters, *A Passion for Excellence*

What Too Often Happens

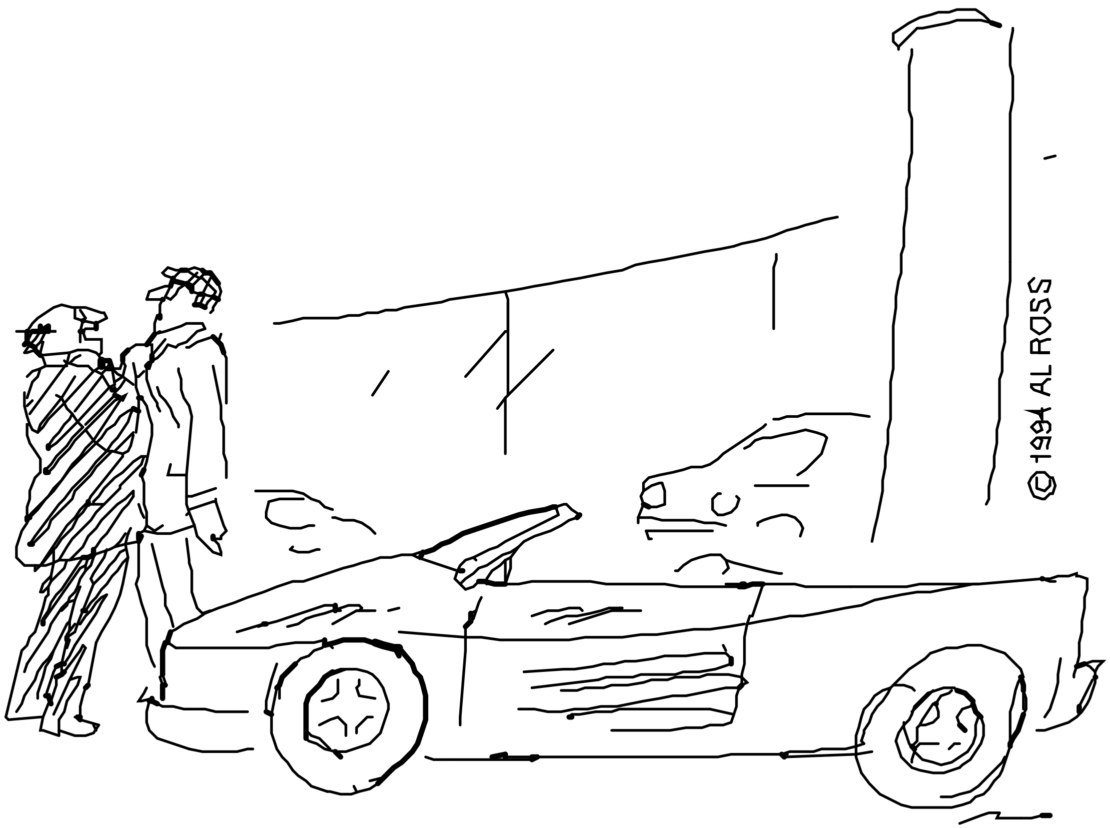
- A (big) process document is written
- The improvement team assumes it is done and deployment is "just give it to the people"
- The process is "deployed"
- The process is ignored, or significant resistance occurs
- The organization gives up or continues to struggle



The Selling Aspect of Getting People to Change

- **What did the sales person do in your best sales experience?**

Individuals Want to be Understood First and Then Have Their Problems Solved



"And I say you can afford it!"

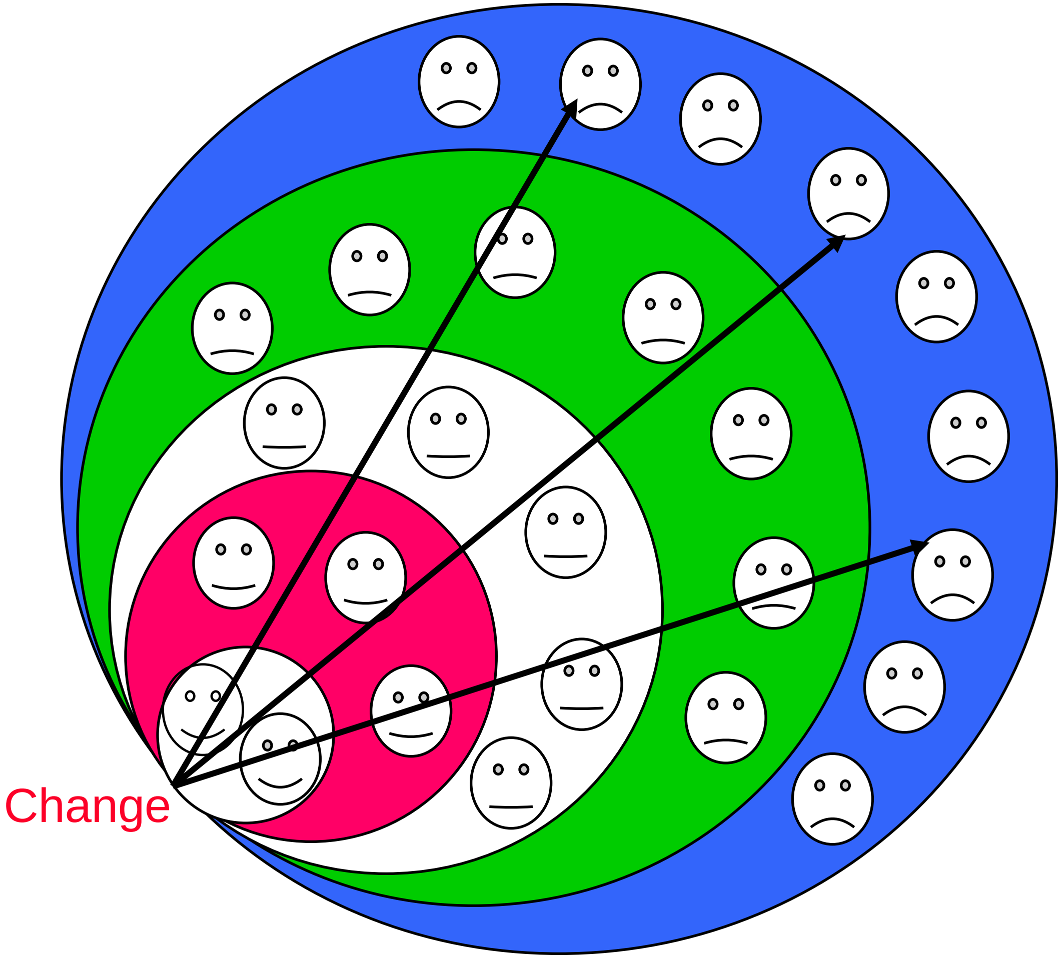
How to Use *Selling*

PROCESS

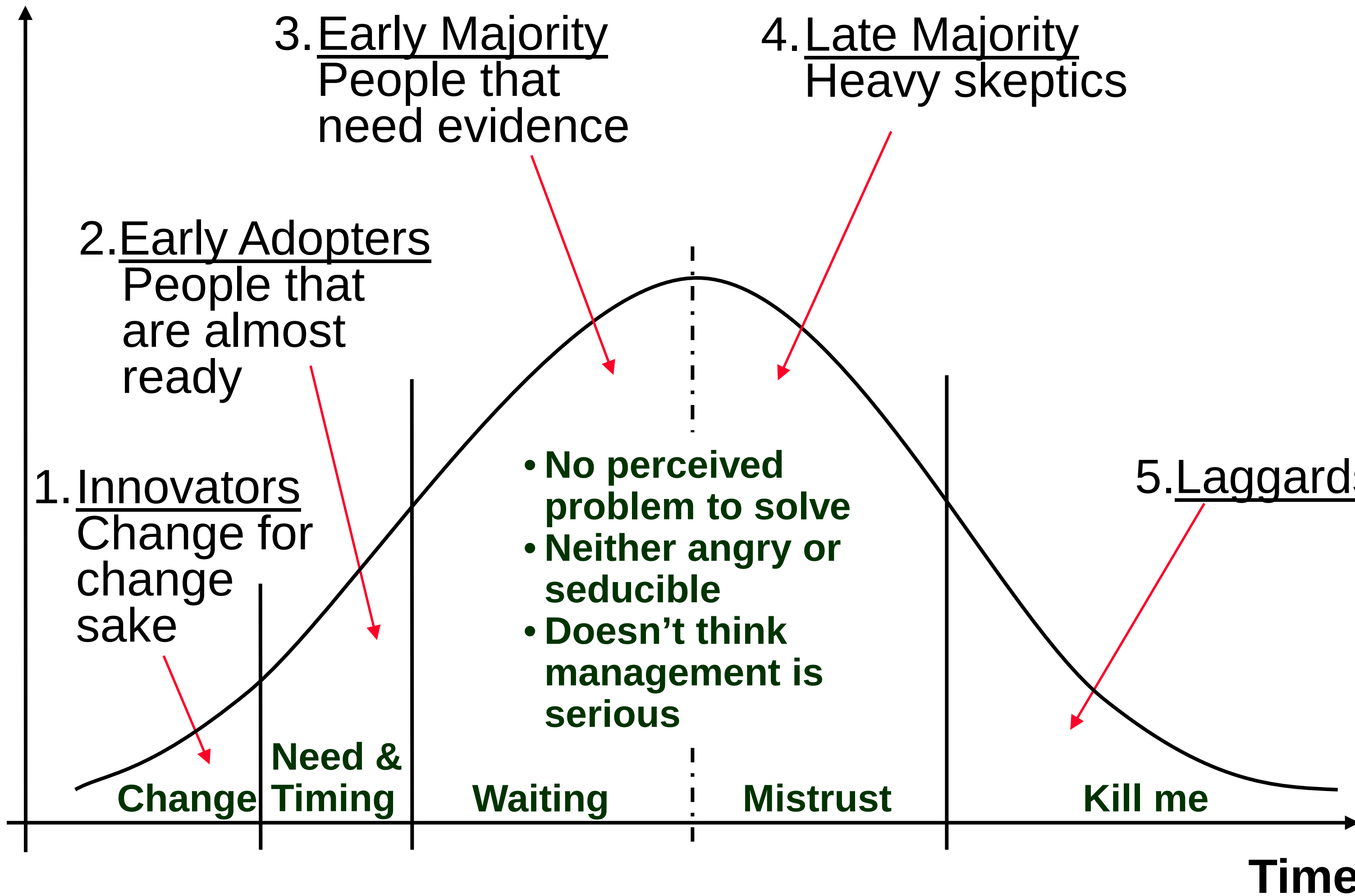
- **Forget** what you are selling
- **Understand** what the customer wants in his/her terms
  - Problems and goals
- Determine the **match** with what you have and what the customer wants
- **Solve** the customer's problem
  - may be a standard or customized solution

Work with the Willing and Needy First

- **A planned and staged approach:**
  - Builds momentum
  - Leverages success stories
  - Provides feedback to refine the solution(s)
  - Easier to manage



What Stages?

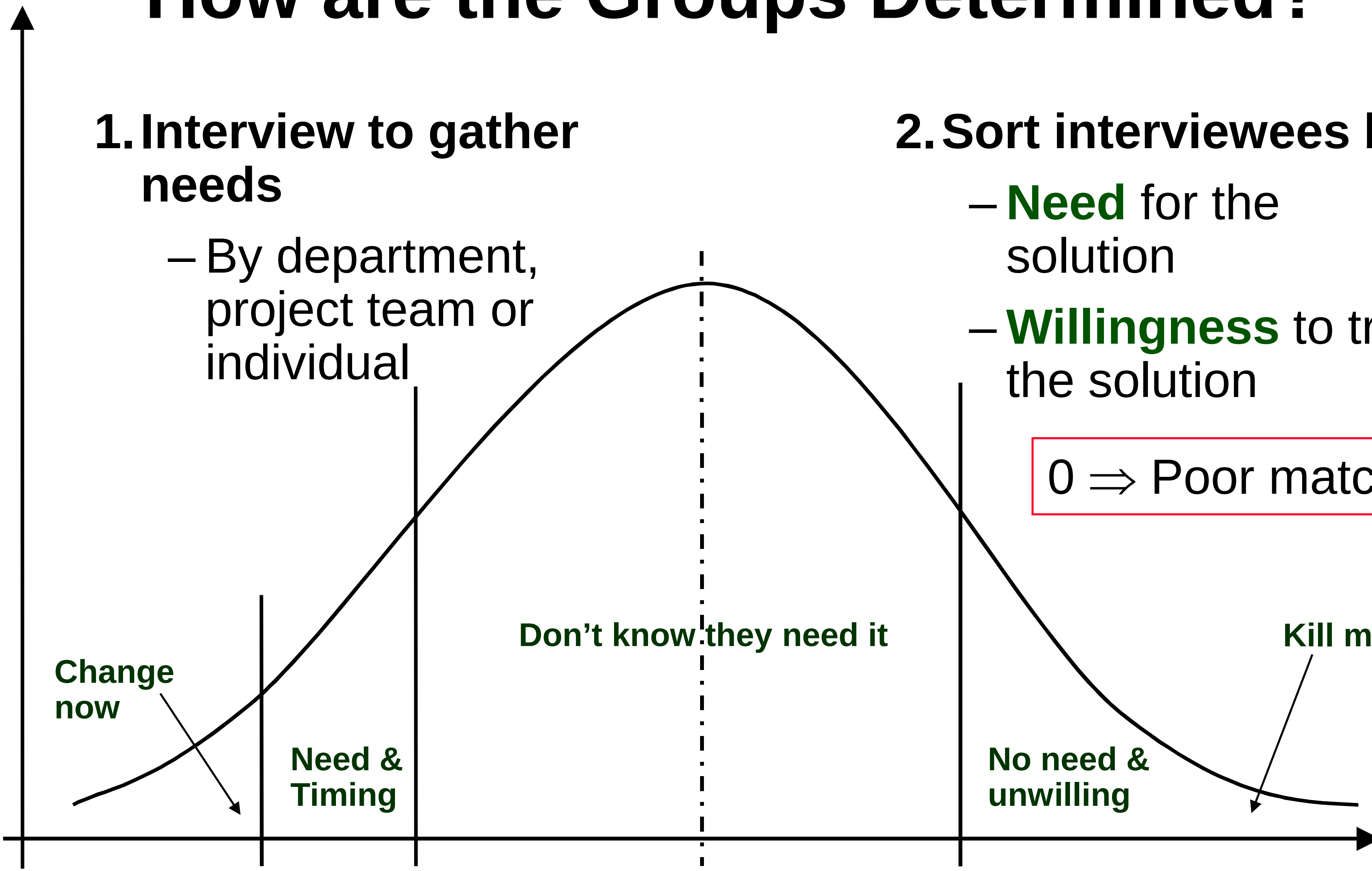


3. Early Majority
People that need evidence
4. Late Majority
Heavy skeptics
2. Early Adopters
People that are almost ready
1. Innovators
Change for change sake
5. Laggards
• No perceived problem to solve
• Neither angry or seducible
• Doesn't think management is serious
Change
Need & Timing
Waiting
Mistrust
Kill me
Time

How are the Groups Determined?

1. **Interview to gather needs**
   – By department, project team or individual
2. **Sort interviewees by**
   – **Need** for the solution
   – **Willingness** to try the solution

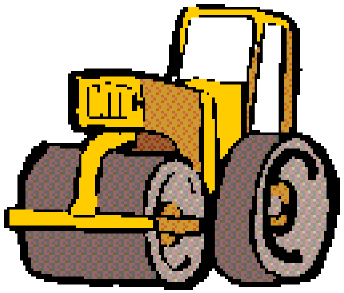
0 ⇒ Poor match



Three Uses of the Adoption Curve

1. Increase the speed of deployment by determining with whom to work and in which order
2. Reduce the risk of failure by building and deploying the solution in increments
3. Determine when to develop a policy and issue an edict

Summary: Implementing the Plan



- Don’t go after the hardest nut (laggard) first
- Focus on real needs (who needs what, when)
- The process provider needs to be flexible and provide appropriate, timely solutions
- PI is not *about* documentation
- Management can lead

Checking Progress

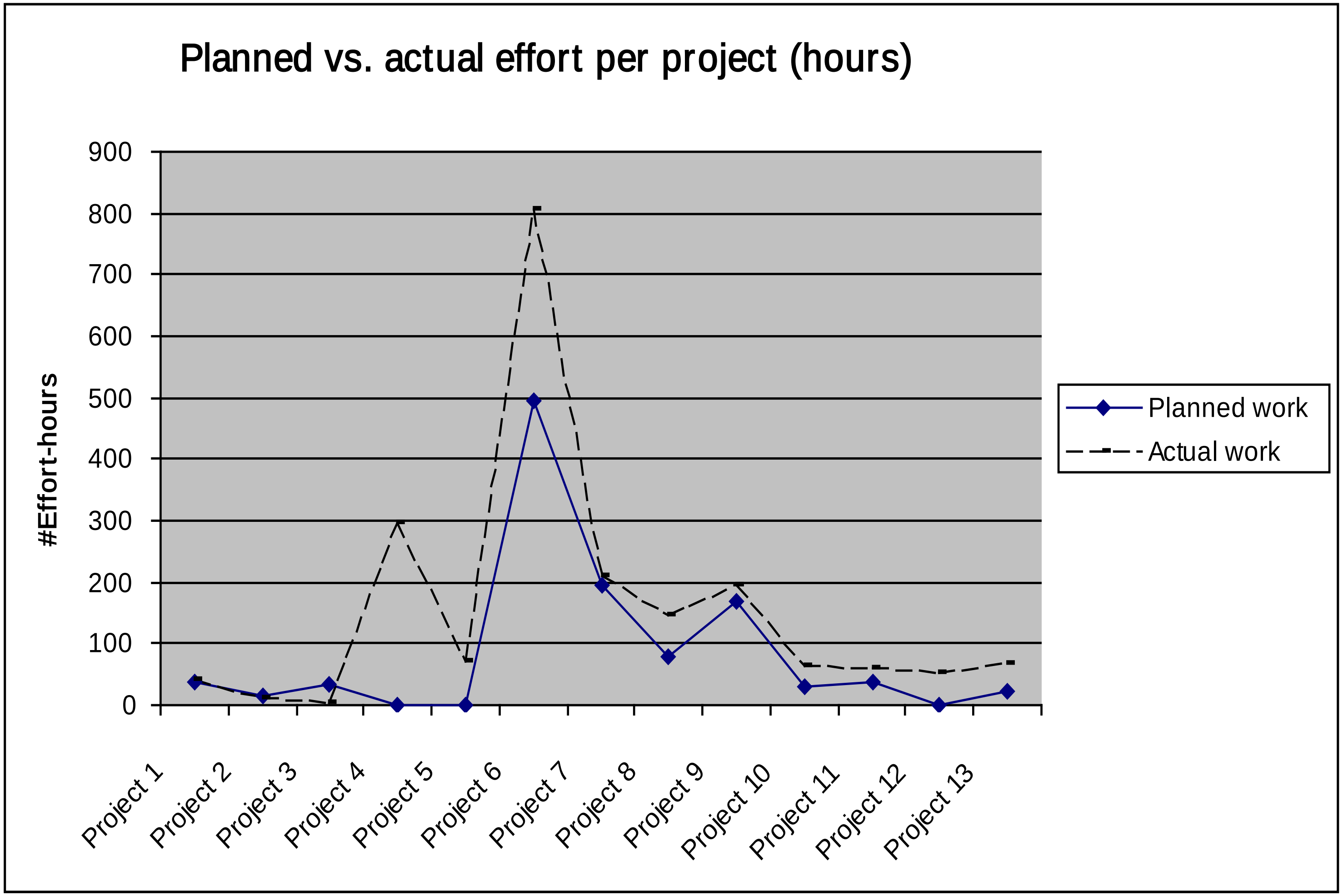
“You can design a measurement system for any conclusion you wish to draw.”

—Gerald Weinberg, *Quality Software Management*

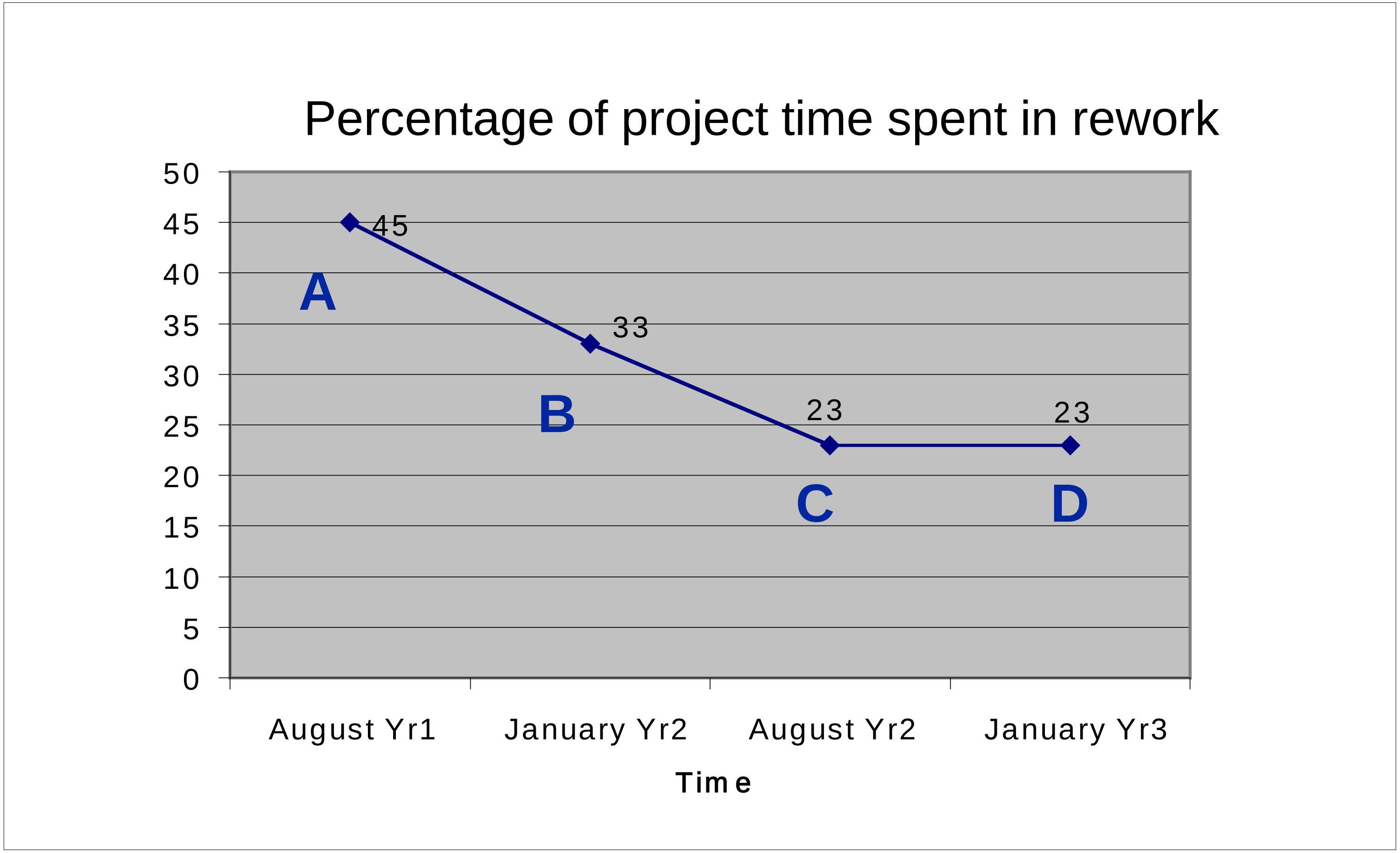
Checking Progress

- Are We Making Progress on the Goals?
- Are We Making Progress on Our Improvement Plan?
- Are We Making Progress on the Improvement Framework?
- What Lessons Have We Learned So Far?

Goal: Meet all Our Cost and Schedule Commitments

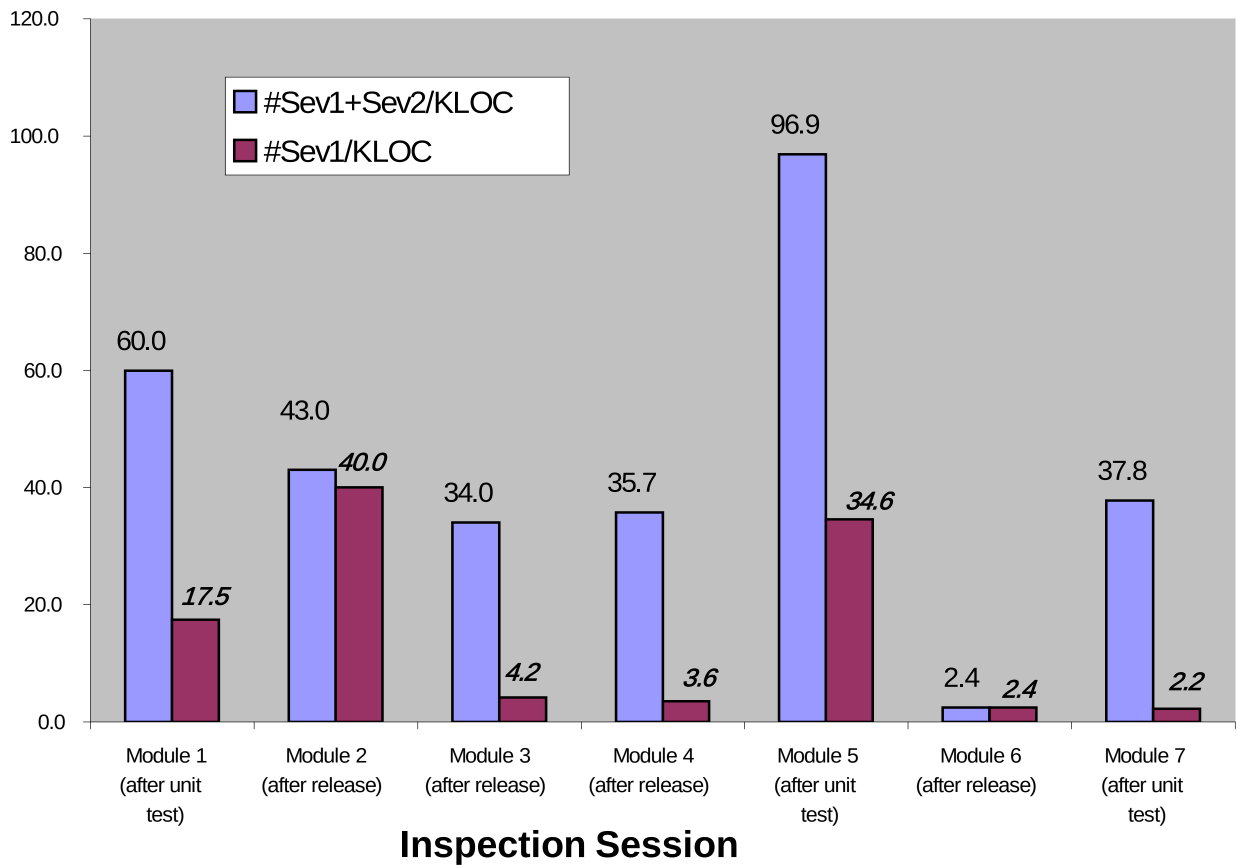


Goal: Reduce Rework to Less Than 20 Percent of Total Project Effort - 1



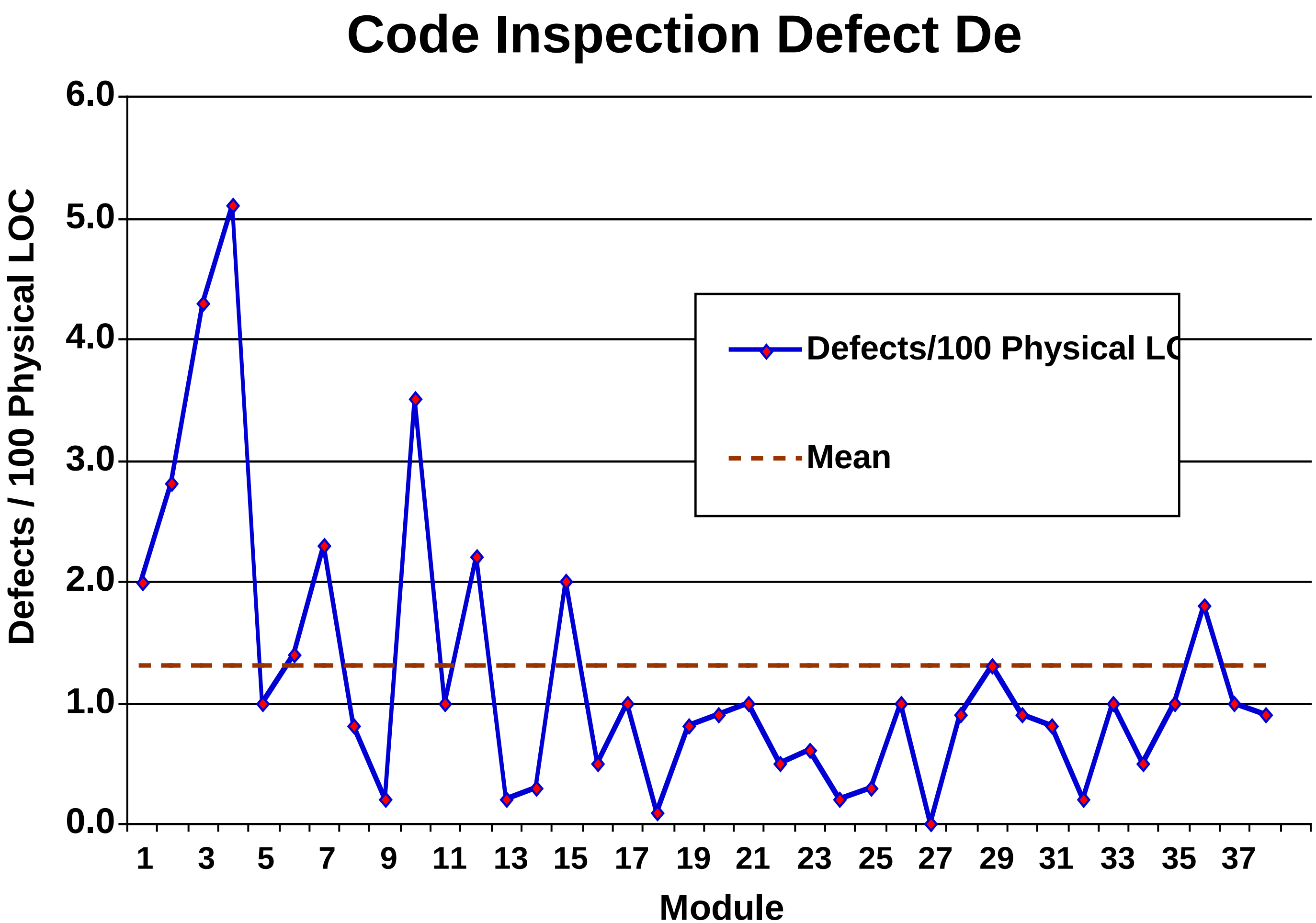
Goal: Reduce Rework to Less Than 20 Percent of Total Project Effort - 2

Java/C++ Inspections – Severity 1 + Severity 2 Defects per Thousands of Lines of Code



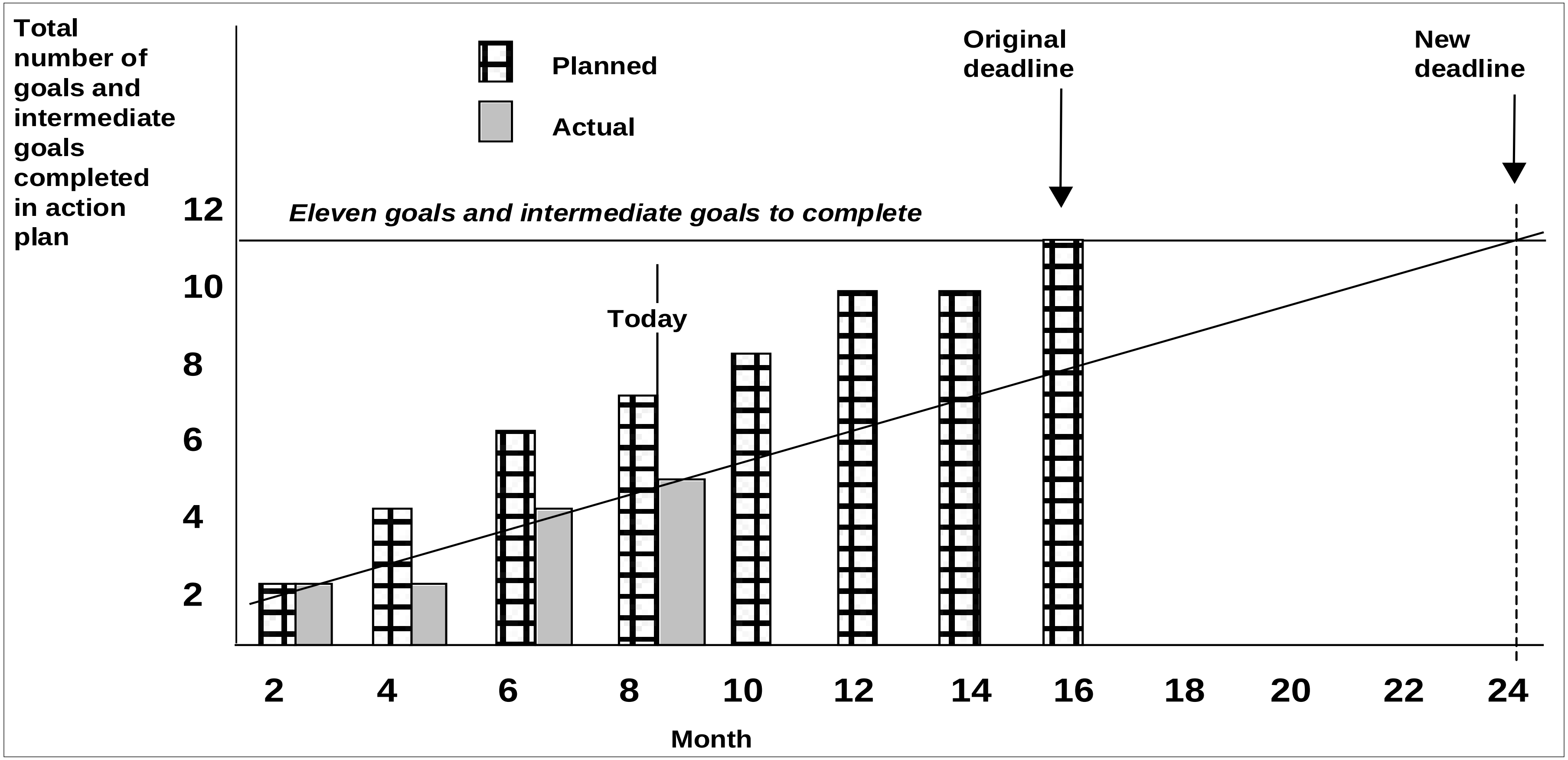
THE PRC GRC

Goal: Reduce Rework to Less Than 20 Percent of Total Project Effort - 3



- Manufacturing control system
- OO/C++
- 167KLOC
- 13 defects/KLOC in code
- 1.38 defects/KLOC in test

Are we Making Progress on Our Improvement Plan?



Trend diagram tracking goal and intermediate goal completion

Are We Making Progress on the Improvement Framework? - 1

Method 1: Count actions that are from the framework

Primary Goal and Intermediate Goals (The results you want)	Purpose of Goal (Why do you want to achieve the goal?)	Actions	Priority (*=essential)
Reduce product development cycle to six to nine months for product X.	**Deliver earlier than competition.**		
Manage changing requirements (based on problem 1).	Prevent schedule slips resulting from expensive scope changes.	Only allow changes to the application interface, not the kernel routines.	1*
		Assign responsibility and authority for performing the REQM process.	2* √
		Check progress and take corrective action .	-
		Improve the library control system to minimize version control errors. Investigate requirements management tools.	3
		Track change requests for the configuration items.	4 √
		Develop an understanding with the requirements providers on the meaning of the requirements .	5 √
		Baseline the requirements before design commences.	6

Are We Making Progress on the Improvement Framework? - 2

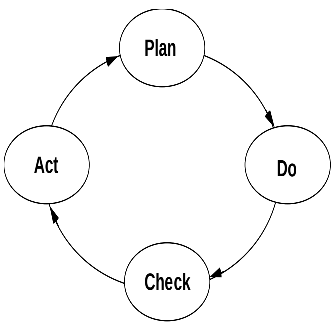
Method 2: Conduct a mini-assessment to establish adoption of practices*

Purpose:

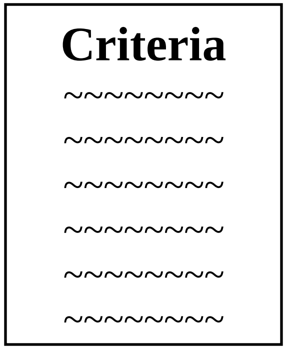
- To **evaluate improvement progress** and make necessary adjustments

Method:

- Develop a **checklist for a verbal interview with each project**
- Conduct interviews with each project (2-3 times per year)

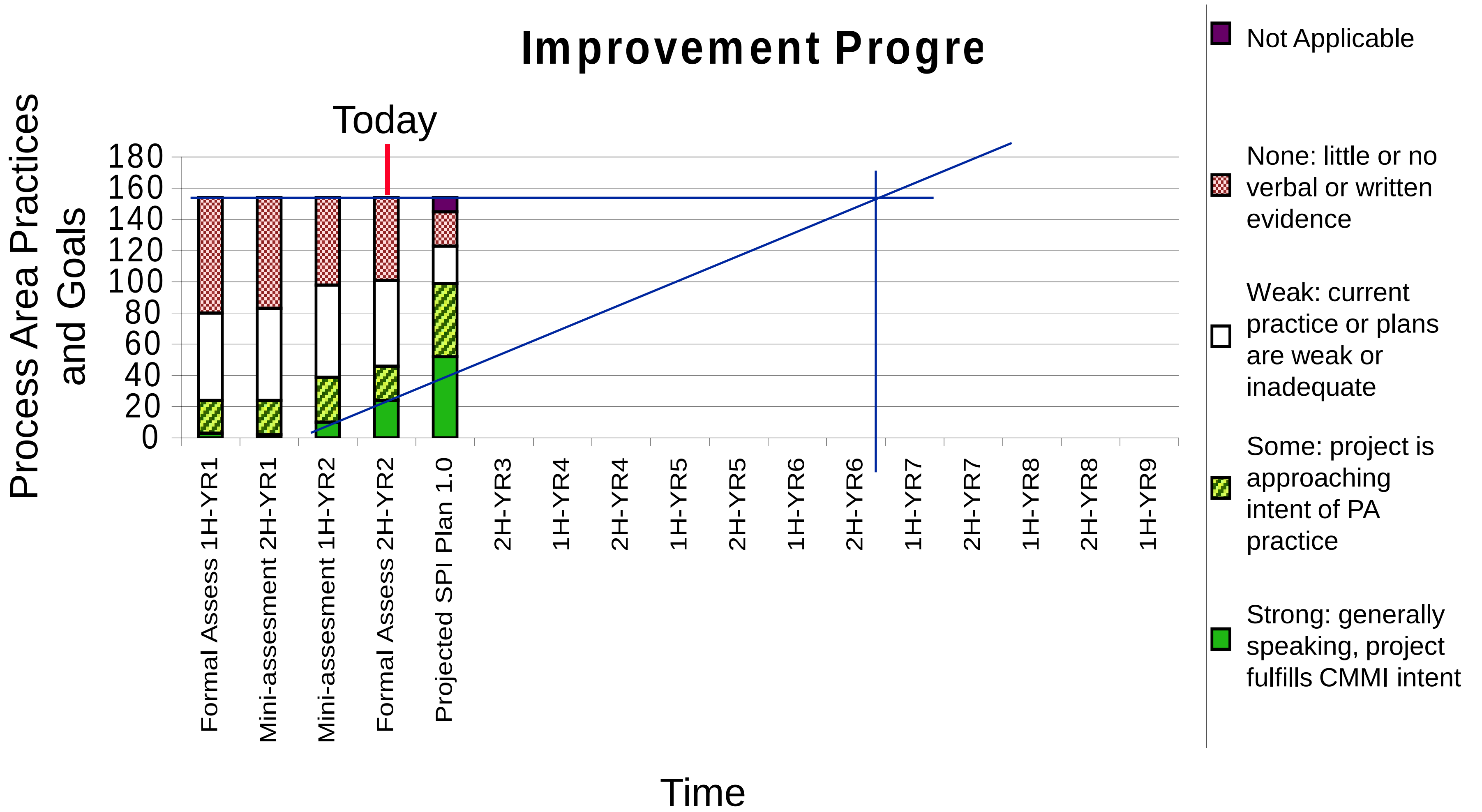


Mini-assessment

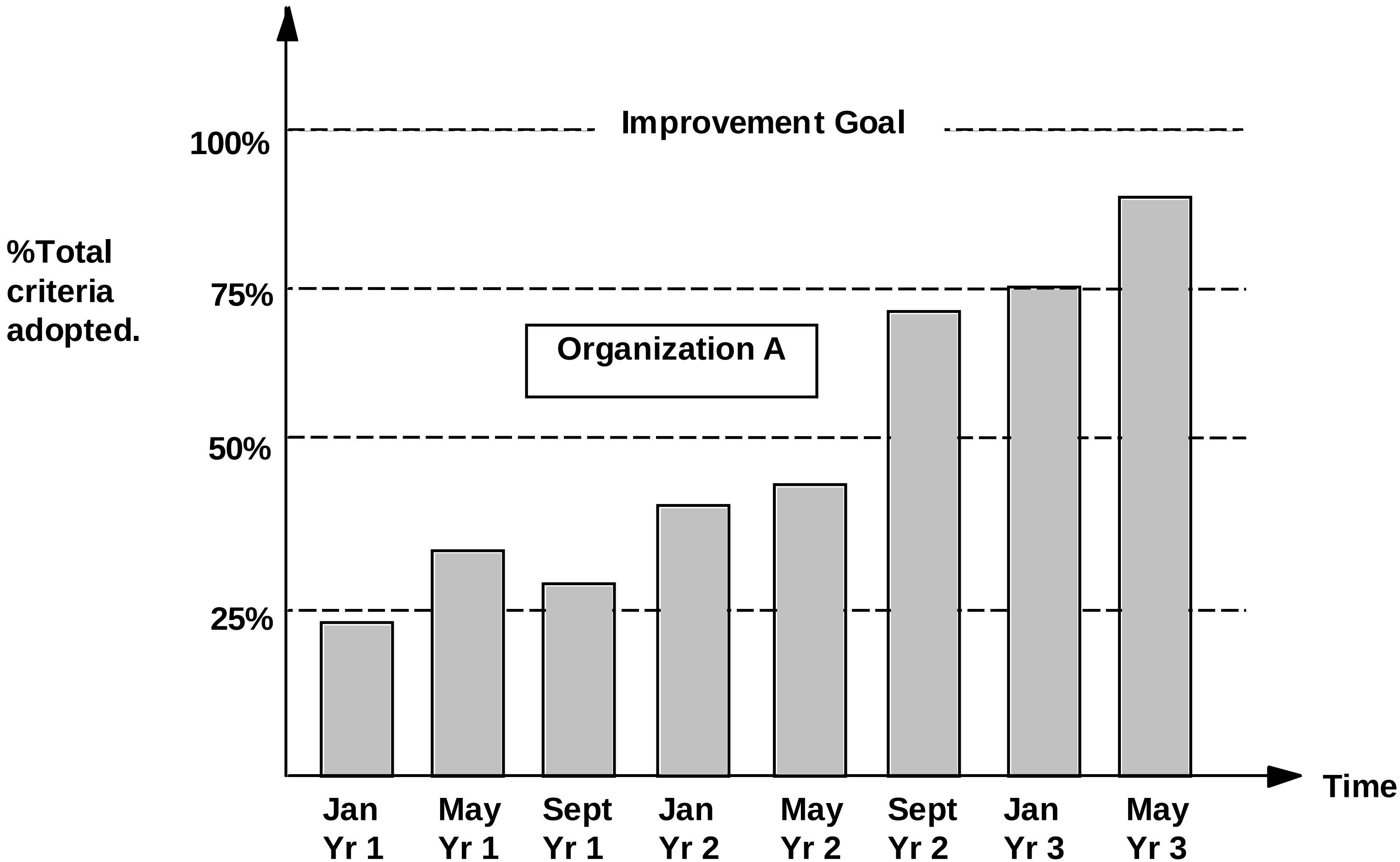


*Potter, N., Sakry, M., "Making Process Improvement Work - A Concise Action Guide for Software Managers and Practitioners," Appendix F. Addison-Wesley, 2002.

Example Mini-assessment Data - 1

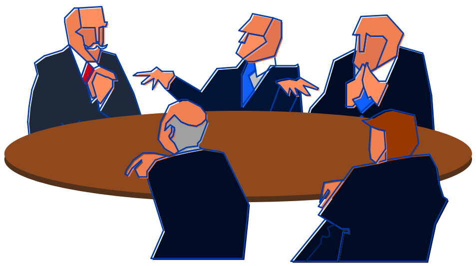


Example Mini-assessment Data - 2



What Lessons Have we Learned so Far?

- **Invite people who are willing to be frank and candid**
  - e.g., PI users, skeptics, managers
- **Select a good objective facilitator**
- **Two hours or less to avoid team fatigue**



Lessons learned agenda

1. **Clarify the scope of the session [10 mins]**
2. **Determine strengths (what went well) [20 mins]**
3. **Determine areas for improvement [30 mins]**
4. **Set priorities [30 mins]**
5. **Determine corrective actions [30 mins]**
   1. Where to use the lesson
   2. Specific corrective actions

Lessons Learned - Strengths

Lesson	Where to Use Lesson
Decentralizing the action plan gives each project team ownership over its plan. Corrective action (CA) = Continue having three separate action plans, one for each of the three product lines.	**Planning**
Don't preach when an example can say everything for you. CA = Have one project each month conduct a one-hour briefing describing the use and benefits of a new technique.	**Implementing**
Guide people in applying each new technique to their work. People have so much going on that they do not know where to start. CA = For each process in the process assets library (PAL), add tailoring guidelines to explain when the process should be used. Provide one-on-one coaching to new project teams.	**Implementing**

Lessons Learned - Improvement Areas

The process-centric approach was very difficult to sell. CA = adopt the goal-problem approach.	**Planning**
Using the same communication technique as everyone else allows the message to be lost. CA = use bright pink 8.5 x 11-inch cards & pizza lunches.	**Implementing**
Allowing private data to become public sets perilous expectations. CA = brief management on new metrics policy.	**Planning**
Be careful of what information you ask for! [Process Assets Library] CA = stop measuring the % of projects that submit to the PAL. Clean out the PAL.	**Planning**
Using a scoring system for process adoption can encourage inappropriate behavior. CA = stop measuring #inspections/year. Re-look at all metrics that can be optimized but lead to little benefit.	**Checking**

Summary - Checking Progress

- Measure what you care about
- Practice measuring
- Lessons-learned data provides additional feedback
- Take corrective action based on what you learn

References

1. Basili, V., and D. Weiss. “A Methodology for Collecting Valid Software Engineering Data.” IEEE Transactions on Software Engineering 1984;SE-10(6):728–738.
2. Block, P., “Flawless Consulting: A Guide to Getting Your Expertise Used.” 2nd ed. San Francisco: Jossey-Bass/Pfeiffer, 1999.
3. 1.1: CMMI Product Development Team. CMMI for Systems Engineering/Software Engineering/Integrated Product and Process Development. Version 1.1 (CMMI-SE/SW/IPPD, v1.1), staged representation. CMU/SEI-2002-TR-004, ESC-TR-2002-004, Pittsburgh: SEI, November 2002.
4. 1.2: CMMI Product Development Team. CMMI for Development. Version 1.2 (CMMI-DEV, v1.2). CMU/SEI-2006-TR-008CMU/SEI-2006-TR-008, ESC-TR-2006-008. August 2006.
5. Dion, R., “Process Improvement and the Corporate Balance Sheet.” IEEE Software 1993:10, no. 4:28–35.
6. Florac, W., and A. Carleton. Measuring the Software Process: Statistical Process Control for Software Process Improvement. Reading, MA: Addison-Wesley, 1999.
7. Grady, R., and D. Caswell. Software Metrics: Establishing a Company-wide Program. Englewood Cliffs, NJ: Prentice-Hall, 1987.
8. Grady, R., Practical Software Metrics for Project Management and Process Improvement. Englewood Cliffs, NJ: Prentice-Hall, 1992.
10. Humphrey, W., “Software Quality Assurance.” In: Managing the Software Process. Reading, MA: Addison-Wesley,1989:137–153.

References

11. Humphrey, W., A Discipline for Software Engineering. Reading, MA: Addison-Wesley, 1995.
12. Moore, G., Crossing the Chasm. New York: Harper-Business, 1991.
13. Carnegie Mellon University/Software Engineering Institute. Edited by: M. Paulk, C. Weber, B. Curtis and M. B. Chrissis. The Capability Maturity Model: Guidelines for Improving the Software Process. Reading, MA: Addison-Wesley, 1995.
14. Robbins A., The Time of Your Life. Audiocassette program. San Diego: Robbins Research International, 1998.
15. Rogers, E., Diffusion of Innovations. New York: The Free Press, 1962.
16. Potter, N., and M. Sakry. "Practical CMM." Software Development 2001;9:65–69.
17. Potter, N., Sakry, M., "Making Process Improvement Work - A Concise Action Guide for Software Managers and Practitioners," Addison-Wesley, 2002.
18. Weinberg, G., The Secrets of Consulting: A Guide to Giving and Getting Advice Successfully. New York: Dorset House Publishing, 1985.
19. Wiegers, K. Software Requirements. Redmond, WA: Microsoft Press, 1999.
20. Yamamura, G., and G. Wigle. "SEI CMM Level 5: For the Right Reasons." CROSSTALK—The Journal of Defense Software Engineering. 1997;10,8:3–8.
21. CMMI: http://www.sei.cmu.edu/cmmi/models/models.html
22. ROI information: http://www.processgroup.com/resources.htm (see ROI Data)

CMMI V1.3 and Architecture

Larry Jones
Mike Konrad

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213-2612

Introductions

Instructor Introduction

Participant Introductions
(mechanics depends on size – individual or show of hands)

- name (if our group is small enough)
- company/position - or type of company (government, defense industry, commercial industry, other)
- background – or job type (manager, technical, process group, other)
- software architecture background / systems architecture background

Tutorial Learning Outcomes

After completing this half-day tutorial, attendees should

- know the importance of architecture to the achievement of business, product, or mission goals
- know that quality attributes have a dominant influence on a system's architecture
- be familiar with essential architecture-centric engineering activities and some example methods
- know how to specify quality attributes meaningfully through scenarios
- be able to identify where architecture-centric activities and work products are described in CMMI V1.3
- appreciate how to interpret the new architecture-centric material in CMMI V1.3
- know where to find out more about architecture-centric engineering practices

Conventions & Caveats for the Tutorial

The coverage of architecture-centric practices in CMMI V1.3 are not restricted to software;

- however, the tutorial providers are most conversant with that domain and thus so is this tutorial.

CMMI V1.3 includes updates to CMMI for Acquisition and CMMI for Services. Our focus in the tutorial will be on CMMI for Development but we will often adopt the shorthand "CMMI V1.3."

CMMI uses the term "product" to refer to what is delivered to the customer or end-user. In this tutorial, we will often use the term "system" to refer to the product.

This tutorial cannot completely convey everything you might like to learn about architecture-centric engineering.

- References are provided at the end for you to learn more.

Expected Background of Participants

Participants must have an understanding of the basics of CMMI models.

- This tutorial is not an introduction to CMMI.
- It is not a substitute for upgrade training.

Familiarity with system and software design is useful, but not required.

Topics to be Covered

CMMI V1.3 – Modern Engineering Practices

Introduction to Architecture

Essential Architecture Practices

Where Are the Architecture-Centric Practices in CMMI V1.3?

Summary

Questions and Answers

There are hands-on exercises to give you a grounding in some key concepts.

Presentation Outline

CMMI V1.3 – Context for modern engineering practices changes

Introduction to Architecture

Essential Architecture Practices

Where Are the Architecture-Centric Practices in CMMI V1.3?

Summary

Questions and Answers

Modern Development Practices in CMMI - The Problem - 1

Much of the engineering content of DEV V1.2 is ten years old.

As DEV was a starting point for the other two constellations, no V1.2 model adequately addresses “modern” engineering approaches.

For example, RD SG 3 and RD SP 3.2 both emphasize functionality and not non-functional requirements (CMMI-SVC SSD SP 1.3 also does too).

Also, Engineering and other PAs rarely mention the following concepts:

- Quality attributes
- Allocation of product capabilities to release increments
- Product lines
- System of systems
- Architecture-centric development practices
- Technology maturation (and obsolescence)
- Agile methods

Modern Development Practices in CMMI - The Problem - 2

The slides that follow portray where we should be today relative to architecture-centric practices – as opposed to how they were portrayed in CMMI V1.2.

Towards the end of today's half-day tutorial, we will revisit how CMMI Version 1.3 addresses these and other modern development practices.

Architecture is Important

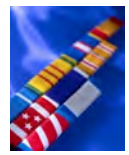
The quality and longevity of a software-reliant system is largely determined by its architecture.

In recent studies by OSD, the National Research Council, NASA, and the NDIA, architectural issues are identified as a systemic cause of software problems in DoD systems.

People are Serious About Architecture

"Software Architect" was identified by CNN Money.com as the #1 "Best Job in America." (Oct 2010)[1]

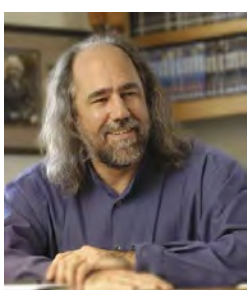
The US Army has mandated that all Program Executive Offices appoint a Chief Software Architect. (May 2009)[2]



1. http://money.cnn.com/magazines/moneymag/bestjobs/2010/snapshots/1.html
2. Memo by LTG N. Ross Thompson, Mil Dept of ASA (ALT) on May 26, 2009.

"Every system has an architecture…

…encompassing the key abstractions and mechanisms that define that system's structure and behavior… In every case - from idioms to mechanisms to architectures - these patterns are either



intentional

or

accidental"

- Grady Booch in the Preface to *Handbook of Software Architecture*

Architecture and Strategy

An *Intentional Architecture* is the embodiment of your business strategy

- Intentional Architecture links technology decisions to business goals

An *Accidental Architecture* limits strategy options

- Accidental Architecture becomes your de facto strategy

Presentation Outline

CMMI V1.3 – Context for modern engineering practices changes

Introduction to Architecture

Essential Architecture Practices

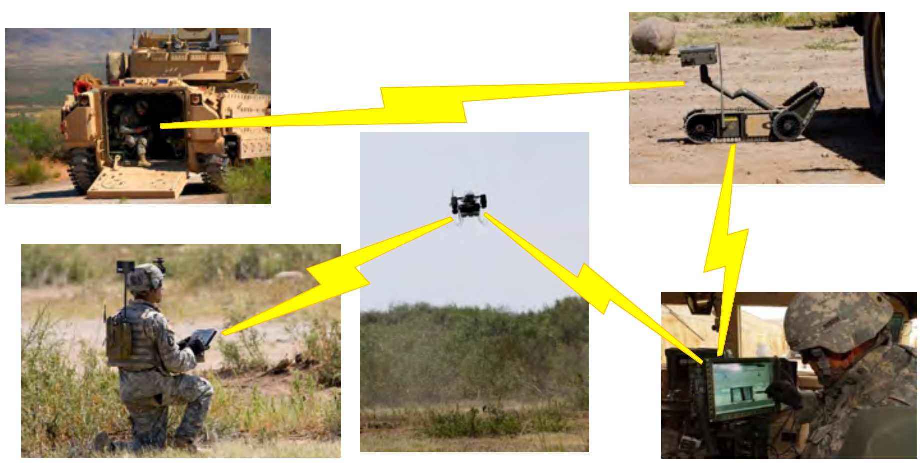
Where Are the Architecture-Centric Practices in CMMI V1.3?

Summary

Questions and Answers

DoD Systems are Increasingly Complex…

…Systems of Systems (SoS) even more so



More and more, software is the integrating element in all manner of systems…

Coping with System/Software Complexity is a Must

2008-2009 Interviews with Army PEOs

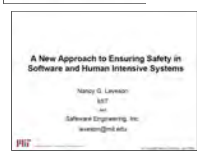
- Relationship between system engineering and software engineering is driving system complexity
- *Example*: Army Software Blocking/Network Capability Sets - decade-long attempt to horizontally integrate Battle Command software across brigade elements

2009 NASA Study

- Software complexity leads to system and operational complexity (and increases risk)

2009 MIT Study

- Software causes systems to be become "interactively complex" (intellectually unmanageable)



Architecture-Centric Practices are Key…

Defense Science Board (1994 & 2000)

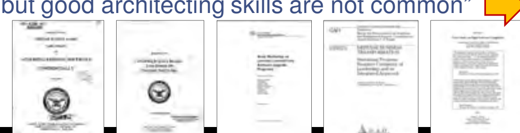
- Software architecture techniques can reduce cost and cycle times
- Architecture is "a central theme for software reuse, product lines, and greater exploitation of commercial technology and practices"

Army Workshop on Weapon Software Upgrade Programs (2001)

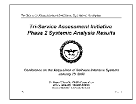
- Architecture is "a key technical focus for the system"
- Architecture is critical in determining the future ability to upgrade the system
- In 2008, GAO testimony noted similar findings for DoD business systems

NASA (2009)

- "Good software architecture is the most important defense against incidental complexity in software designs, but good architecting skills are not common"

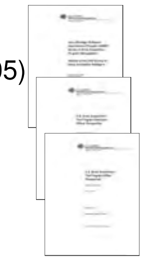


…But Practices Haven't Kept Up



DoD Tri-Service Assessment Initiative (2003)

- Review of 21 DoD program assessments
  - poor software architecture practices are one of the *systemic causal factors* of software-reliant systems issues



SEI surveys and interviews of Army PMs and PEOs (2004 & 2005)

- PMs/PEOs felt prime contractors' software architecture abilities were only about average
  - Yet, they also felt government program office staffs were not sufficiently skilled to evaluate software architectures

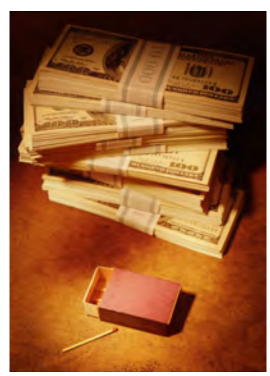
SEI analysis of results from 18 architecture evaluations (2006)

- >50% of the programs had significant *program* risks driven by lack of architecture training/tools and poor architecture planning
- ~2/3 of risks discovered were risks of *omission*
  - *e.g.,* architectural decisions either not made or not captured

Fixing this Sounds Expensive!

Compared to what?

- Over-committing because you don't have a blueprint for the whole system?
- Inefficiency from inability to coordinate work?
- Late rework when defects found in test and integration?
- Delivering late and over budget?
- Developing a failed product that doesn't meet stakeholder's needs?



Architecture is About Structure and Decisions

Structures result from decisions

- Business / mission goals provide a reasoned basis for decisions.
- Each decision is a tradeoff that enables something and precludes other things.
- Tradeoffs are driven by quality attribute requirements.

This is true regardless of the domain – commercial or defense.

Class Exercise 1

Value Proposition for Architecture-Centric Engineering

VP **Architecture-centric engineering enables the ongoing cost-effective achievement of system-related business and mission goals.**

- Early identification and mitigation of design risks result in fewer downstream, costly problems and cost savings in integration and test.
- Sound structure analyses provide objective confidence for achieving system quality.
- Predictable system quality supports the achievement of business and mission goals, which translates into competitive advantage.
- Appropriate flexibility enables cost-effective system evolution.

Why Is Software Architecture Important?

Represents ***earliest*** design decisions
- hardest to change
- most critical to get right
- communication vehicle among stakeholders

First design artifact addressing
- performance
- modifiability
- reliability
- security

Key to systematic ***reuse***
- transferable, reusable abstraction

Key to system ***evolution***
- manage future uncertainty
- assure cost-effective agility

The **right architecture** paves the way for system **success**.
The **wrong architecture** usually spells some form of **disaster**.

Software Architecture and Development and Acquisition Risk

Risk mitigation early in the life cycle is key.

- The software architecture is an early life cycle artifact.
- Mid-course correction is possible before great investment.
- Risks don't become problems that have to be addressed during integration and test.

Agile Architecture = Responsiveness

Architecture-centric engineering and an agile development approach are *not* at odds.

Agile development approaches enable you to

- Take on large projects and initiatives
- Break them into smaller chunks (iterations)
- Manage risk
  - Execute-Learn-Feedback-Improve

Agile Architecture provides the blueprint for your iterations

- Enable efficient incremental development
- Minimize technical debt
- Early analysis of qualities like performance and availability
- Efficiently address global qualities like security

Common Symptoms Stemming From Architectural Deficiencies

Operational

- Communication bottlenecks under various load conditions in a system or throughout a system of systems (SoS)
- Systems that hang up or crash; portions that need rebooting too often
- Difficulty synching up after periods of disconnect and resume operations
- Judgment by users that system is unusable for variety of reasons
- Database access sluggish and unpredictable

Developmental

- Integration schedule blown, difficulty identifying root causes of problems
- Proliferation of patches and workarounds during integration and test
- Integration of new capabilities taking longer than expected, triggering breaking points for various resources
- Significant operational problems ensuing despite passage of integration and test
- Anticipated reuse benefits not being realized

Sample Issues Detectable From Architectural Decisions

Availability:

- Having a single point of failure
- Having no availability mechanisms
- Using an infrastructure that does not support availability mechanisms

Performance:

- Not knowing performance requirements
- Failure to meet performance requirements
  - Not performing any performance modeling or prototyping
  - Unfamiliarity with infrastructure choices
  - Not using known performance mechanisms

Security:

- No support for security
- Not using known mechanisms to support security goals

Modifiability:

- Allocating functionality in a way that jeopardizes portability
- Not supporting the addition and deletion of different devices
- Lack of attention to potential growth paths

Integration:

- Problems with migrating legacy systems
- Lack of uniformity in key areas

This is What Happens

without careful architectural design.

And so it is with software.

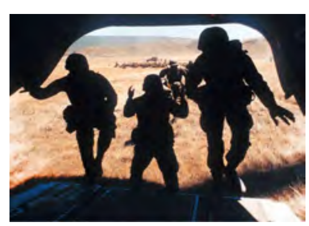
Without Effective Software Architecture Practices

…. you get poorly designed software architectures.

Poorly designed software architectures result in

- Greatly inflated integration and test costs
- Inability to sustain systems in a timely and affordable way
- Lack of system robustness
- Undesired, disparate behaviors at the system and at the system-of-systems levels
- In the worst case, product or project cancellation
- In all cases, failure to best support the war fighter

Failure to Meet Business and Mission Goals



A Warning (PERMISSION REQUESTED)

"Architecture" is a very overloaded word.

- All the good words are taken.
- We will explain some common uses of the term and how they differ.

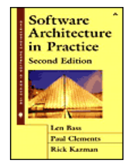
What Is A Software Architecture?

Informally, software architecture is the blueprint describing the software structure of a system.

Formal Definition

"*The* ***software architecture*** *of a program or computing system is the* ***structure or structures*** *of the system, which comprise the* ***software elements, the externally visible properties*** *of those elements, and the* ***relationships among*** *them.*"[1]

[1] Bass, L.; Clements; P. & Kazman, R. *Software Architecture in Practice, Second Edition.* Boston, MA: Addison-Wesley, 2003.



Implications of Our Definition

Software architecture is an abstraction of a system.

Software architecture defines the properties of elements.

Systems can and do have many structures.

Every software-intensive system *has* an architecture.

Just having an architecture is different from having an architecture that is known to everyone.

If you don't develop an architecture, you will get one anyway –
and you might not like what you get!

Structures and Views - 1

One house, many views

- Carpentry view
- Plumbing view
- Electrical view
- Ductwork view

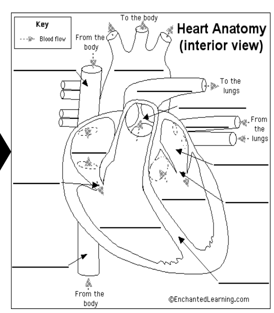
No single view accurately represents the house.

No single view can be used to build the house.

Although these views are pictured differently, and each has different properties, all are related. Together, they describe the architecture of the house.

Structures and Views - 2

A human body comprises multiple *structures*.



a *static view* of one human *structure*

a *dynamic view* of that *structure*

One body has many structures, and those structures have many views. So it is with software.

Enterprise Architecture

Enterprise architecture is a means for describing business structures and the processes that connect them.[1]

- Describes the flow of information and activities between various groups within the enterprise that accomplish some overall business activity

Software and its design are not typically addressed explicitly in an enterprise architecture.

[1] Zachman, John A., *"A Framework for Information Systems Architecture." IBM Systems Journal*, 26, 3 (1987): 276-292.

System Architecture

A ***system architecture*** describes the elements and interactions of a complete system including its hardware elements and its software elements.

System Architecture: "*The fundamental and unifying system structure defined in terms of system elements, interfaces, processes, constraints, and behaviors.*"[1]

Systems Engineering is a design and management discipline useful in designing and building large, complex, and interdisciplinary systems.[2]

[1] Rechtin, E. *Systems Architecting: Creating and Building Complex Systems.* Englewood Cliffs, NJ : Prentice-Hall, 1991.

[2] International Council On Systems Engineering (INCOSE), Systems Architecture Working Group, 1996.

Where Does Software Architecture Fit?

Enterprise architecture and system architecture provide an environment in which software lives.

- Both provide requirements and constraints to which software architecture must adhere.
- Both are affected by the properties of the software architecture.
- Elements of both are likely to contain software architecture.
- **Neither substitutes for or obviates a software architecture.**

There is a mutual influence and interaction between software, system, and enterprise architectures.

In a large, complex, software-reliant system **both software and system architectures are critical** for ensuring that the system meets its business and mission goals.

What About System of Systems?

Each software-intensive system in a system of systems (SoS) has system and software architectures.

The system of systems has an architecture where the elements are themselves the software architectures of the individual systems.

Software architecture is even more important in an SoS context, not less.

Does DoDAF Address Software Architecture?

Unfortunately, no.

- DoDAF views are required
- software architecture views are not

The Department of Defense Architecture Framework (DoDAF) describes an "architecture" for a large-scale system or system-of-systems.

DoDAF uses the concept of views of a system

- operational view (OV) – participant relationships and information needs
- system (SV) – relates capabilities and characteristics to operational requirements
- technical (TV) – prescribes standards and conventions
- all (AV)

DoDAF views were developed for different purposes and do not address software architecture.

Presentation Outline

CMMI V1.3 – Context for modern engineering practices changes

Introduction to Architecture

Essential Architecture Practices

Where Are the Architecture-Centric Practices in CMMI V1.3?

Summary

Questions and Answers

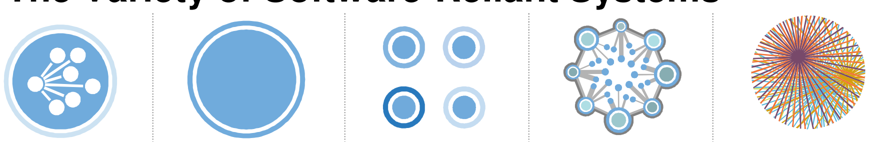
What is Architecture-Centric Engineering?

Architecture-Centric Engineering (ACE) is the discipline of using architecture as the focal point for performing ongoing analyses to gain increasing levels of confidence that systems will support their missions.

Architecture is of enduring importance *because it is the right abstraction for performing ongoing analyses throughout a system's lifetime.*

The **SEI ACE Initiative** develops principles, methods, foundations, techniques, tools, and materials in support of creating, fostering, and stimulating widespread transition of the ACE discipline.

The Variety of Software-Reliant Systems



Embedded systems
software embedded in hardware devices

Stand-alone systems
software applications

Software product lines
families of similar systems

Systems of systems
federations of independent systems

Ultra-large-scale systems
webs of software-reliant systems, people, economies, and cultures

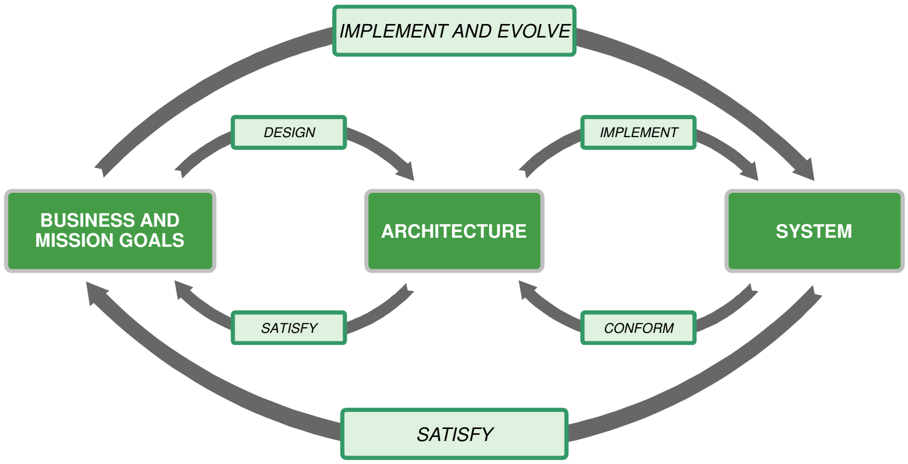
There are interactions among these types of systems.
The behavior of all these systems is largely determined by their structure.

Architecture-centric engineering addresses all types and scales of systems.

Predict and control behavior **Assure and bound behavior**

Coupling to organizational structure and practices increases

ACE Design and Analysis



Principles of ACE

1. Regardless of scale, architecture is the **appropriate abstraction** for reasoning about business/mission goal satisfaction.
2. **Quality attributes** have a dominant influence on a system's architecture.
3. Architectural prescriptions must be demonstrably satisfied by the **implementation**.

Architecture – A Bridge to Goal Satisfaction

A good architectural representation should have

- sufficient detail to reason about mission and business goal satisfaction
- sufficient abstraction for a relatively small number of architects to conceptually understand the system
- sufficient detail to appropriately constrain implementation.

All design involves tradeoffs.

Lacking mission and business drivers, the architect has to make assumptions about priorities.

Given well-stated mission and business drivers, the architect has a basis for knowing the priorities among tradeoffs.

Software Engineering Institute | Carnegie Mellon CMMI V1.3 and Architecture 47

Principles of ACE

1. Regardless of scale, architecture is the **appropriate abstraction** for reasoning about business/mission goal satisfaction.
2. **Quality attributes** have a dominant influence on a system's architecture.
3. Architectural prescriptions must be demonstrably satisfied by the **implementation**.

Software Engineering Institute | Carnegie Mellon CMMI V1.3 and Architecture 48

Software System Development

Functional Software Requirements

If function were all that mattered, any monolithic software would do, ..***but other things matter***…

The important quality attributes and their characterizations are key.

- Modifiability
- Interoperability
- Availability
- Security
- Predictability
- Portability

analysis, design, development, evolution

Quality Attribute Drivers → **Software Architecture** → **Software**

has these qualities

The Non-functional Requirements

Quality Attribute Requirements

Quality attributes include

- Performance
- Availability
- Interoperability
- Modifiability
- Usability
- Security
- Etc.

Quality attribute requirements stem from business and mission goals.

Key quality attributes need to be characterized in a system-specific way.

Otherwise, they are not operational.

Users Need Both Functions and Qualities

Required capability
Low learning threshold
Ease of use
Predictable behavior
Dependable service
Timely response
Timely throughput
Protection from unintended intruders and viruses
……
Software system/mission goals should address user needs.
User needs often translate to quality attribute requirements.
Scenarios are a powerful way to characterize quality attributes and represent user and other stakeholder views.

Specifying Quality Attributes

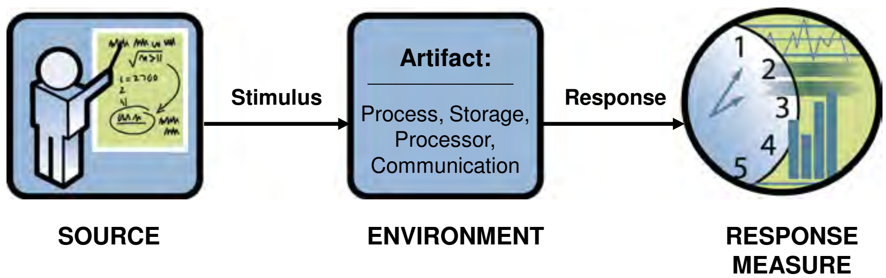
Quality attributes are rarely captured *effectively* in requirements specifications; they are often vaguely understood and weakly articulated.

Just citing the desired qualities is not enough; it is meaningless to say that the system shall be “modifiable” or “interoperable” or “secure” without details about the context.

The practice of specifying quality attribute scenarios can remove this imprecision and allows desired qualities to be evaluated meaningfully.

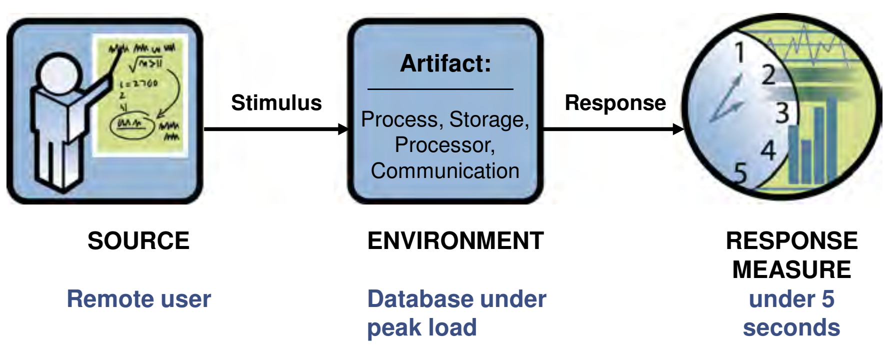
A quality attribute scenario is a short description of an interaction between a stakeholder and a system and the response from the system.

Parts of a Quality Attribute Scenario



Example Quality Attribute Scenario

A "performance" scenario: A remote user requests a data base report under peak load and receives it in under 5 seconds.



Class Exercise 2

Principles of ACE

1. Regardless of scale, architecture is the **appropriate abstraction** for reasoning about business/mission goal satisfaction.

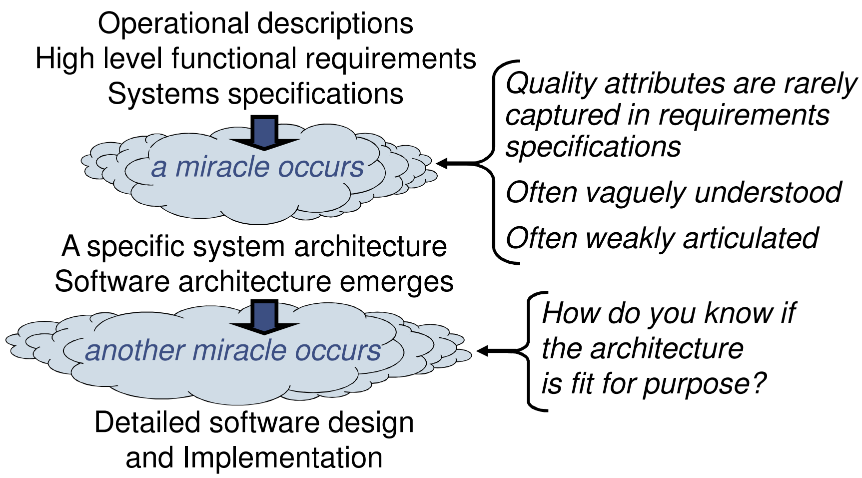
2. **Quality attributes** have a dominant influence on a system's architecture.
   - Quality attribute requirements stem from business and mission goals.
   - Key quality attributes need to be characterized in a system-specific way.
   - Scenarios are a powerful way to characterize quality attributes and represent stakeholder views.

3. Architectural prescriptions must be demonstrably satisfied by the **implementation**.

Principles of ACE

1. Regardless of scale, architecture is the **appropriate abstraction** for reasoning about business/mission goal satisfaction.
2. **Quality attributes** have a dominant influence on a system's architecture.
3. Architectural prescriptions must be demonstrably satisfied by the **implementation**.

Typical Software Development Paradigm



Architecture-Centric Activities

Architecture-centric activities include the following:

- creating the **business case** for the system
- understanding the **requirements**
- **creating and/or selecting** the architecture
- **documenting and communicating** the architecture
- **analyzing or evaluating** the architecture
- **implementing** the system based on the architecture
- ensuring that the implementation **conforms** to the architecture
- **evolving** the architecture so that it **continues to meet business and mission goals**

Some SEI Techniques, Methods, and Tools

creating the **business case** for the system	
understanding the **requirements**	*Quality Attribute Workshop (QAW) Mission Thread Workshop (MTW)*
creating and/or selecting the architecture	*Attribute-Driven Design (ADD) and ArchE*
documenting and communicating the architecture	*Views and Beyond Approach; AADL*
analyzing or evaluating the architecture	*Architecture Tradeoff Analysis Method (ATAM); SoS Arch Eval; Cost Benefit Analysis Method (CBAM); AADL*
implementing the system based on the architecture	
ensuring that the implementation **conforms** to the architecture	*ARMIN*
evolving the architecture so that it **continues to meet business and mission goals**	*Architecture Improvement Workshop (AIW) and ArchE*
ensuring **use of effective architecture practices**	*Architecture Competence Assessment*

Building the Business Case for the System

How to do this is beyond the scope of this tutorial.

Some common business / mission drivers for systems include

- Reduce total cost of ownership
- Improve capability/quality of system
- Improve market position
- Support improved business processes
- Improve confidence in and perception of system

Results gleaned from

- 25 architecture evaluations
  - 18 government systems, 7 commercial systems
- 190 distinct business goals

Kazman & Bass, *Categorizing Business Goals for Software Architectures*, CMU/SEI-2005-TR-021
http://www.sei.cmu.edu/reports/05tr021.pdf

Understanding the Requirements – The SEI's Quality Attribute Workshop

The purpose of the SEI Quality Attribute Workshop (QAW) is to discover, early in the life cycle, the driving quality attribute requirements of a software-intensive system.

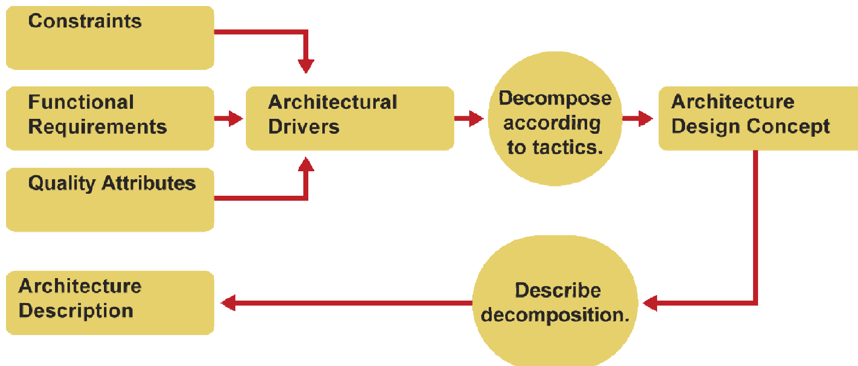
QAW Steps

1. QAW Presentation and Introductions
2. Business/Programmatic Presentation
3. Architectural Plan Presentation
4. Identification of Architectural Drivers
5. Scenario Brainstorming
6. Scenario Consolidation
7. Scenario Prioritization
8. Scenario Refinement

Barbacci, et al., *Quality Attribute Workshops (3rd Ed.),* CMU/SEI-2003-TR-016
http://www.sei.cmu.edu/library/abstracts/reports/03tr016.cfm

An Approach to Architecture Creation

The Attribute-Driven Design (ADD) method is an approach to defining a software architecture by basing the design process on the quality attribute requirements of the system.



Class Exercise 3

Creating the Architecture

How to do this is beyond the scope of this tutorial.

Part of the ADD approach is to pick architectural *patterns* and *tactics* that address particular quality attributes.

Patterns represent a packaging of a number of design decisions we refer to as *tactics*.

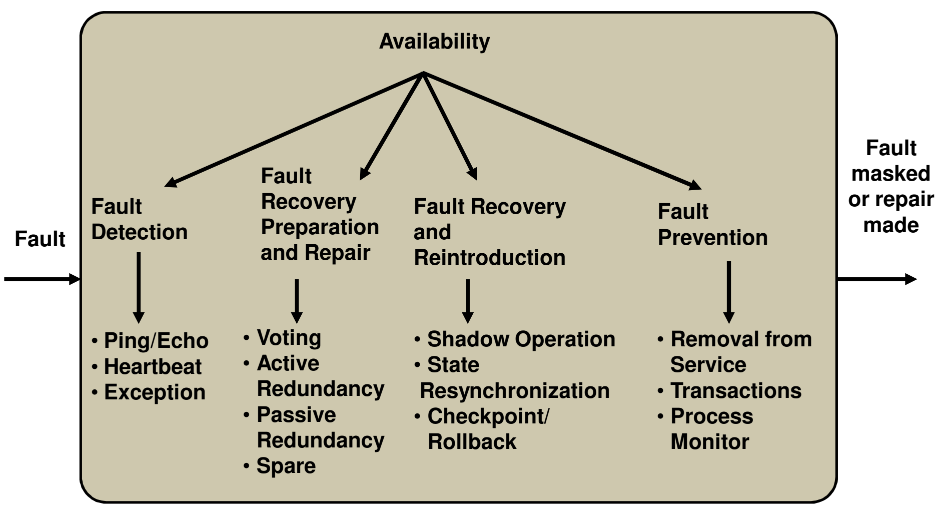
Each *tactic* is a design option available to the architect.

A pattern typically employs several different tactics to promote various quality attributes.

Example: Tactics to influence *availability* (keep faults from becoming errors) include

- Fault Detection
- Fault Recovery
- Fault Prevention

Summary of Availability Tactics



Other Tactics

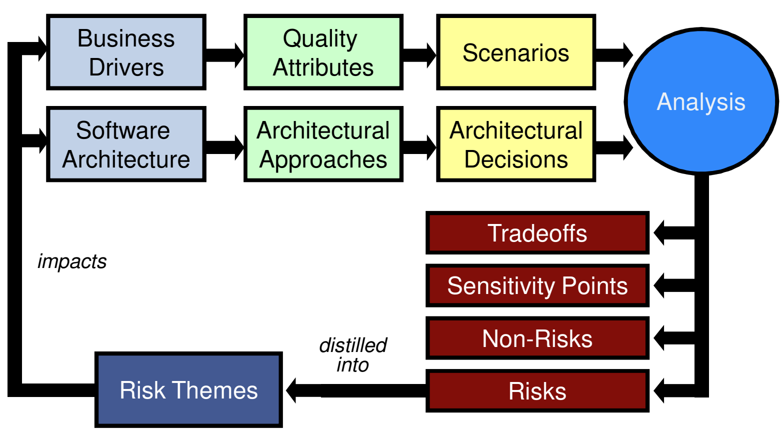
There are tactics for

- modifiability
- performance
- security
- testability
- usability

See *Software Architecture in Practice* for a more complete treatment of the subject.

Analyzing the Architecture – SEI's Architecture Tradeoff Analysis Method® (ATAM®)

The ATAM is an architecture evaluation method that focuses on multiple quality attributes.



ATAM Phases

ATAM evaluations are conducted in four phases.

Phase 0: Partnership and Preparation	→	Phase 1: Initial Evaluation	→	Phase 2: Complete Evaluation	→	Phase 3: Follow-Up

Duration: varies
Meeting: primarily phone, email

Duration: 1.5 - 2 days each for Phase 1 and Phase 2
Meeting: typically conducted at customer site

Duration: varies
Meeting: primarily phone, email

ATAM Evaluative Phases (1 & 2)

1. Present the ATAM
2. Present business drivers
3. Present architecture

Presentation

4. Identify architectural approaches
5. Generate quality attribute utility tree
6. Analyze architectural approaches

Investigation and Analysis

7. Brainstorm and prioritize scenarios
8. Analyze architectural approaches

Testing

9. Present results

Reporting

Phase 1

Phase 2 = Recap of Phase 1 plus

Documenting the Software Architecture

Architecture documentation establishes the set of design decisions that must be made along the way to establishing and maintaining the architecture.

An architecture is a multidimensional construct, too involved to be seen all at once.

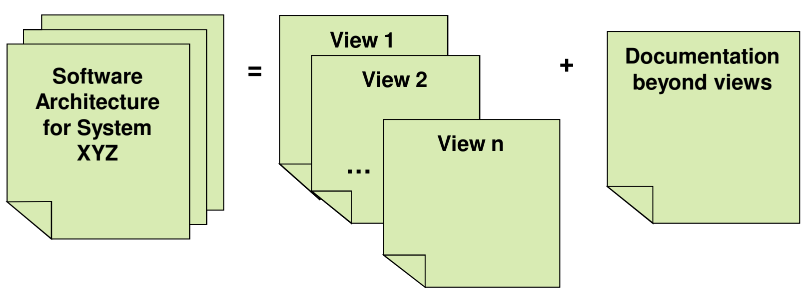
Recall: systems are composed of many structures.

A view is a representation of a structure.

We use views to manage complexity by separating concerns.

View-Based Documentation

Views give us our basic principle of architecture documentation



Documenting an architecture is a matter of documenting the relevant views, and then adding documentation that applies to more than one view.

The choice of views used depends on the nature of the system and the stakeholder needs.

Software Architecture Documentation Needs

Runtime views to show how software will handle:

- hazards, faults, and errors
- fault tolerance/reconfigurations
- performance
- data (e.g., quality, timeliness, ownership, access privileges)
- interface boundaries

Non-runtime views of software (vital to project planning, allocating work assignments, designing for modifiability, reusability, portability, extensibility, etc., facilitating incremental development, and a host of other critical purposes)

Architectural decisions and the rationale/implications/impact of those decisions on key system qualities

So How Well Does This Work?
Study: Impact of Army Architecture Evaluations

Twelve Army programs that had conducted ATAM or QAW exercises in a study to elicit the perceived impact the ATAM evaluations and QAWs had on system quality and the practices of the acquisition organization.

Results showed

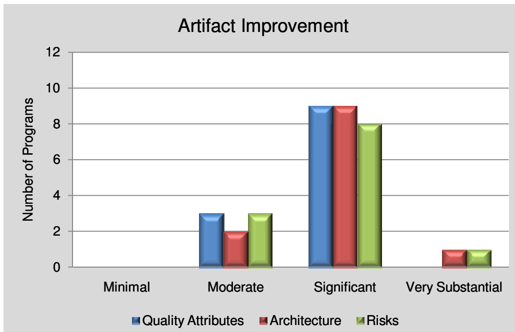
- 6/12: cost less than or equal to traditional techniques
- 10/12: quality of results greater than or equal to traditional techniques
- 10/12: helped understand and control cost and schedule
- 12/12: increased understanding of system's quality attribute requirements, design decisions, and risks
- 12/12: good mechanism for communication among stakeholders
- 8/12: improved the architecture

The context of use had a significant impact on the results enjoyed.

Architecture-centric acquisition is key to reaping maximal benefit.

Architecture Practices are Having an Impact 1 of 2

Results of 2008 survey of 12 Army projects that employed ATAM/QAW[2]



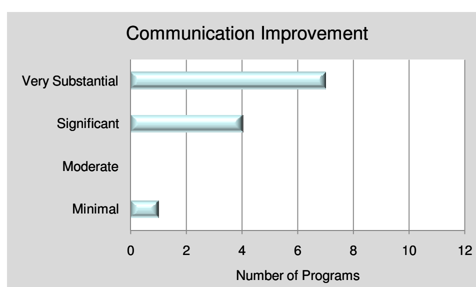
- Most reported *significant* improvement in their architecturally-significant artifacts
- Architecture teams were able to achieve understanding of stakeholder expectations and the implications of architectural decisions on user needs

[2] Source: Impact of Army Architecture Evaluations, CMU/SEI-2009-SR-007

Architecture Practices are Having an Impact 2 of 2

Results of 2008 survey of 12 Army projects that employed ATAM/QAW

- Majority reported *very substantial* or *significant* improvement in stakeholder communication
- Stakeholders, collectively, are able to achieve a common understanding of the system under development
  - Increases likelihood that product will address expectations/user needs
  - Improves chances for program success



Themes From the Army Presentations - 1

"The ATAM architecture evaluations resulted in improved documentation, improved communication, reduced risk in schedule and cost, and a higher quality product to the warfighter."

"Independent, 3rd party architecture evaluation is quite beneficial for programs that are considered high risk, and/or for which the PM has no visibility into architecture/design."

"The ATAM is an effective mechanism for getting the stakeholders to work together and identify architectural risks early in the acquisition/development life cycle when they can still be mitigated in a cost effective manner."

- "It is important that programs (and their supporting contractors) have good risk management procedures so that risks uncovered by an ATAM evaluation are properly tracked and mitigated."

Themes From the Army Presentations - 2

"QAW should be part of the operational architecture community to ensure quality attributes, and not just functionality, are appropriately addressed."

- "QAW results were very beneficial to conducting follow-on ATAM evaluations because the QAW scenarios and architectural drivers can carry forward."
- "QAWs at the system and system of system (SoS) requirements levels are a good thing and should especially be applied on US Joint Forces Command (JFCOM) programs so all stakeholder requirements can be suitably addressed."

"QAWs and the ATAM are making a very good impact on Army programs, perhaps more than the SEI is aware of. The SEI needs to codify this and send the message to Army management."

"The importance of having had the backing of Army senior leadership and ASSIP funding is that the beneficiaries— the Army programs—went from "Nay-Sayers" to "Yea-Sayers.""

Implementing and checking conformance

Press on to implementing the system in accordance with the architecture.

Have processes and supporting tools to check for conformance with the architecture.

Unfortunately, a lot of this work today is not automated.

Principles of ACE

1. Regardless of scale, architecture is the **appropriate abstraction** for reasoning about business/mission goal satisfaction.

2. **Quality attributes** have a dominant influence on a system's architecture.

3. Architectural prescriptions must be demonstrably satisfied by the **implementation**.
   - Software architecture must be central to software development activities.
   - These activities must have an explicit focus on quality attributes.
   - These activities must directly involve stakeholders – not just the architecture team.
   - The architecture must be descriptive and prescriptive.

Extending these ideas to Systems and Systems of Systems

The previous discussion was based largely on software engineering practices.

The ideas and techniques have been extended into the realm of systems and systems-of-systems.

Initial results are positive.

System / SoS Architecture Problems

Severe integration and runtime problems arise due to inconsistencies in how quality attributes are addressed in system and software architectures.

This is further exacerbated in an SoS context where major system and software elements are developed concurrently and oftentimes independently.

A uniform approach for **specifying** quality attribute requirements and **evaluating SoS and system architectures** against such requirements is needed.

The Need for Augmented Mission Threads in DoD SoS Architecture Definition

DoDAF is the SoS architecture framework for the DoD.

- It provides a good set of architectural views for an SoS architecture.
- It inadequately addresses cross-cutting quality attribute considerations.

System use cases focus on a functional slice of the system.

More than DoDAF and system use cases are needed to ensure that the SoS architecture satisfies its end-to-end functional requirements *and* quality attribute needs.

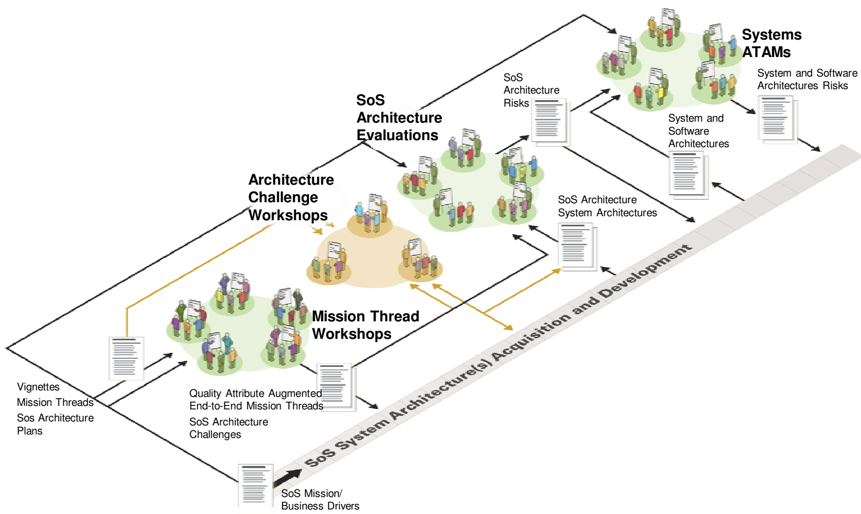
SoS end-to-end mission (operational or user) threads augmented with quality attribute considerations are needed to help develop, and later evaluate, the SoS architecture.

One Approach

SEI developed and applied a two-pronged approach to address the early identification of quality attribute inconsistencies, ambiguities, and omissions within system and SoS architectures (in Directed and Acknowledged SoS contexts).

1. Perform a "first pass" identification of inconsistencies, ambiguities, and omissions across the constituent systems, at the SoS level, using end-to-end mission threads that are augmented with quality attribute concerns from SoS stakeholders.
   The approach involves a series of workshop and evaluations.
   - Mission Thread Workshop
   - Architecture Challenge Workshop
   - SoS Architecture Evaluation
2. Constituent systems that are “problematic” are further evaluated using the system and software architecture evaluation method (based on the ATAM), using the augmented mission threads from the Mission Thread Workshops.
   - System and Software ATAM

SoS and Quality Attribute Elicitation, Specification, and Analysis



Architectural Reuse

An architecture represents a significant investment.

Why use it for only one system?

Most organizations produce families of similar systems, differentiated by features.

The DoD acquires families of similar systems.

The Real Truth About Reuse

Reuse means using an item more than once.

"*The XYZ System is built with 80% reuse.*"

A statement like this is vacuous.

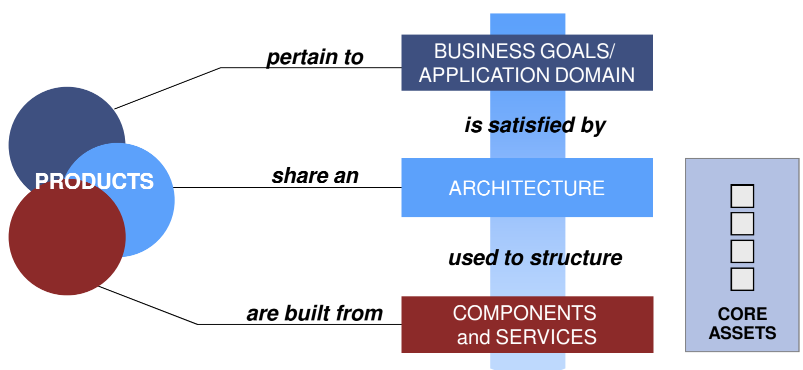
- It is not clear what is being reused.
- It is not clear that the "reuse" has any benefit.

Reusing code or components without an architecture focus and without pre-planning results in

- Short-term perceived win
- Long-term costs and problems
- Failure to meet business goals

Software Engineering Institute | Carnegie Mellon CMMI V1.3 and Architecture 87

Reuse That Pays Off: Software Product Lines



Product lines
- take economic advantage of commonality
- bound variation

Software Engineering Institute | Carnegie Mellon CMMI V1.3 and Architecture 88

Software Product Lines

A software product line is a ***set*** of software-intensive systems sharing a ***common, managed set of features*** that satisfy the specific needs of a ***particular market segment or mission*** and that are ***developed from a common set of core assets*** in a ***prescribed way.***

How Do Product Lines Help?

Product lines amortize the investment in these and other *core assets:*

- requirements and requirements analysis
- domain model
- software architecture and design
- performance engineering
- documentation
- test plans, test cases, and test data
- people: their knowledge and skills
- processes, methods, and tools
- budgets, schedules, and work plans
- components and services

TOTAL LIFE CYCLE REUSE ⇒ MORE BENEFIT

PRODUCT LINES = STRATEGIC REUSE

Successful Software Product Lines

Improvements in cost, time to market, and productivity that come with successful product lines abound.

- **Cummins** reduced the time it takes to produce software for a diesel engine from one year to one week.
- **Motorola** realized a 400% productivity improvement in a family of one-way pagers.
- **Hewlett-Packard** reduced time to market by a factor of seven and increased productivity by a factor of four in a family of printers.
- The **NRO** built a ground control system with 10% of the expected number of developers and reduced defects by 90%.
- **Nokia** reports producing 25 to 30 different phone models per year by using a product line approach.

Widespread Application - 1

akvasmart
Feed control and farm management software

BOEING
Bold Stroke Avionics

E-COM Technology Ltd.
Medical imaging workstations

hp invent
Firmware for computer peripherals

Lucent Technologies Bell Labs Innovations
5ESS telecommunications switch

ABB Asea Brown Boveri
Gas turbines, train control, semantic graphics framework

Dialect
Internet payment gateway infrastructure products

ERICSSON
AXE family of telecommunications switches

LG
Elevator control systems

NOKIA
Mobile phones, mobile browsers, telecom products for public, private and cellular networks

AXIS COMMUNICATIONS
Computer printer servers, storage servers, network camera and scanner servers

DNV
Customized solutions for transportation industries

GM
Software for engines, transmissions and controllers

LSI LOGIC
RAID controller firmware for disk storage units

NASA
Interferometer product line

Widespread Application - 2

PHILIPS
High-end televisions,
PKI telecommunications switching system,
diagnostic imaging equipment

RICOH
Office appliances

BOSCH
Automotive gasoline systems

Rockwell Collins
Commercial flight control system avionics,
Common Army Avionics System (CAAS),
U.S. Army helicopters

SALiON TARGET. WIN. DELIVER.
Revenue acquisition
management systems

SIEMENS
Software for viewing and quantifying
radiological images

TELVENT
Industrial supervisory control
and business process
management systems

testo
Climate and flue gas
measurement devices

symbian
EPOC operating system

alltel **FIDELITY** NATIONAL FINANCIAL
Support software

NAVSEA WARFARE CENTERS
Test range facilities

U.S. ARMY
Command and control
simulator for Army fire
support

MOTOROLA
Pagers product line

Software Product Lines in the DoD

Organizations having or adopting a software product line approach include

- US Army C-E LCMC: Advanced Multiplex Test System (AMTS)
- Army Training Information Systems Directorate: Army Training Information Architecture (ATIA)
- Overwatch Textron Systems: Overwatch Intelligence Center (OIC) Software Product Line
- OneSAF: OneSAF Product Line Architecture
- Joint Battle Command – Platform product line
- Rockwell Collins: Common Avionics Architecture System (CAAS)
- PEO Simulation, Training & Instrumentation (PEO STRI): Live Training Transformation Components plus Common Training Instrumentation Architecture (LT2/CTIA)
- PEO Simulation, Training & Instrumentation (PEO STRI): SE Core - Synthetic Environment Core (SE Core) is the Army's Common Virtual Environment (CVE)
- US Army Joint Fires Product Line
- Common Driver Training Product Line
- Northrop Grumman Common Link Integration Processing product line
- USMC Live Training Transformation product line

Presentation Outline

CMMI V1.3 – Context for modern engineering practices changes

Introduction to Architecture

Essential Architecture Practices

Where Are the Architecture-Centric Practices in CMMI V1.3?

Summary

Questions and Answers

Modern Development Practices in CMMI - 1

For Version 1.3, CMMI provides better guidance in support of architecture-centric practices

- creating the **business case** for the system (partially in RD)
- understanding the **requirements** (RD)
- **creating and/or selecting** the architecture (TS)
- **documenting and communicating** the architecture (RD, TS)
- **analyzing or evaluating** the architecture (RD, TS, VAL, VER)
- **implementing** the system based on the architecture (TS; A/PL notes)
- ensuring that the implementation **conforms** to the architecture (VER)
- **evolving** the architecture so that it **continues to meet business and mission goals** (implicit in the phrase "establish and maintain")

The above repeats the "Architecture-Centric Activities" slide seen earlier. (Elaborations indicate where the practice is addressed in CMMI V1.3.)

Modern Development Practices in CMMI - 2

CMMI V1.3 provides improved terminology to support architecture-centric practices

- Updated the glossary to include new terms (and modified some old terms)
- Updated the informative material (especially ARD and ATM in ACQ; RD, TS, and VER in DEV; and SSD in SVC) to:
  - make use of the new terms
  - bring more emphasis to quality attributes and thus strike a better balance between functional and non-functional requirements
- Replaced selected uses of overloaded terms such as "performance" with an appropriate qualifying phrase.

CMMI Support for: creating the business case for the system

CMMI V1.3 touches on the "why" for the business in many places, including OPF, OPM, OPP, QPM, RD. Focusing here only on RD:

RD SP 1.1 Elicit Needs

Elicit stakeholder needs, expectations, constraints, and interfaces for all phases of the product lifecycle.

RD SP 1.2 Transform Stakeholder Needs into Customer Requirements

Transform stakeholder needs, expectations, constraints, and interfaces into prioritized customer requirements.

[snip] Relevant stakeholders representing all phases of the product's lifecycle should include *business* as well as technical functions. In this way, *concepts* for all product related lifecycle processes are considered concurrently with the *concepts* for the products. *Customer requirements result from informed decisions on the business as well as technical effects of their requirements. [Emphasis added]*

CMMI Support for: understanding requirements - 1

CMMI support for understanding requirements is mostly found in the RD PA (and secondarily in a few other places, especially VAL).

SG 1 Develop Customer Requirements

SP 1.1 Elicit Needs

SP 1.2 ~~Develop the~~Transform Stakeholder Needs into Customer Requirements

SG 2 Develop Product Requirements

SP 2.1 Establish Product and Product Component Requirements

SP 2.2 Allocate Product Component Requirements

SP 2.3 Identify Interface Requirements

SG 3 Analyze and Validate Requirements

SP 3.1 Establish Operational Concepts and Scenarios

SP 3.2 Establish a Definition of Required Functionality and Quality Attributes

SP 3.3 Analyze Requirements

SP 3.4 Analyze Requirements to Achieve Balance

SP 3.5 Validate Requirements

CMMI Support for: understanding requirements - 2

Specific Goal and Practice Changes (most of them in RD)

Changed RD SG 3 so it no longer appears to focus on functionality.

SG 3 Analyze and Validate Requirements

The requirements are analyzed and validated~~, and a definition of required functionality is developed~~.

Changed SP 1.2 to make stakeholder/customer priorities more explicit.

SP 1.2 Transform Stakeholder Needs into ~~Develop the~~ Customer Requirements

Transform stakeholder needs, expectations, constraints, and interfaces into prioritized customer requirements.

Changed RD SP 3.2 to add emphasis to non-functional requirements.

SP 3.2 Establish a Definition of Required Functionality and Quality Attributes

Establish and maintain a definition of required functionality and quality attributes.

CMMI Support for: understanding requirements - 3

RD (especially) and other PAs: Informative Material Changes

Added and revised the informative material throughout these PAs to appropriately mention the following engineering concepts:

- quality attributes (i.e., non-functional requirements or "ilities")
- product lines, system of systems
- architecture-centric practices
- allocation of product capabilities to release increments
- technology maturation (and obsolescence)

These concepts are mentioned in example boxes, in examples provided in the notes, and in discussion that mentions various approaches that can be used.

When functional requirements are discussed, mention of quality attributes is added to balance the view of requirements.

CMMI Support for: understanding requirements - 4

In RD SP 1.1 Elicit Needs

- Added the following examples of techniques to elicit needs:
  - o [snip] Questionnaires, interviews, and scenarios (operational ~~scenarios~~, sustainment, and development) obtained from end users
  - o Operational, sustainment, and development walkthroughs and end-user task analysis
  - o Quality attribute elicitation workshops with stakeholders
- Added Example Work Product:

  Results of requirements elicitation activities

In RD SP 1.2 Transform Stakeholder Needs into Customer Requirements

- Added the following new subpractice:

  2. Establish and maintain a prioritization of customer functional and quality attribute requirements.

CMMI Support for: understanding requirements - 5

In RD SP 2.1 Establish Product and Product Component Requirements

- Added a note to Subpractice 2 (deriving requirements that result from design decisions):

  Architectural decisions, such as selection of architecture patterns, introduce additional derived requirements for product components. For example, the Layers Pattern will constrain dependencies between certain product components.

- Added the following new subpractice:

  3. Develop architectural requirements capturing critical quality attributes and quality attribute measures necessary for establishing the product architecture and design.

CMMI Support for: understanding requirements - 6

In RD SP 2.2 Allocate Product Component Requirements

- Added a note:

  The product architecture provides the basis for allocating product requirements to product components. [snip] In cases where a higher level requirement specifies ~~performance~~ a quality attribute that will be the responsibility of more than one product component, the ~~performance must~~quality attribute can sometimes be partitioned for unique allocation to each product component as a derived requirement, however, other times the shared requirement should instead be allocated directly to the architecture. [snip]

- Revised first four subpractices:

  1. Allocate requirements to functions.
  2. Allocate requirements to product components and the architecture.
  3. Allocate design constraints to product components and the architecture.
  4. Allocate requirements to delivery increments.

CMMI Support for: understanding requirements - 7

In RD SG 3 Analyze and Validate Requirements

- Added a note:

  Architecturally significant quality attributes are identified based on mission and business drivers.

In RD SP 3.1 Establish Operational Concepts and Scenarios

- Changed Subpractice 1 to read:

  1. Develop operational concepts and scenarios that include ~~functionality, performance~~operations, installation, development, maintenance, support, and disposal as appropriate.

     Identify and develop scenarios, consistent with the level of detail in the stakeholder needs, expectations, and constraints in which the proposed product or product component is expected to operate.

     Augment scenarios with quality attribute considerations for the functions (or other logical entities) described in the scenario.

CMMI Support for: understanding requirements - 8

In RD SP 3.2 Establish a Definition of Required Functionality and Quality Attributes

- Added a note (split here for readability):

  Such approaches have evolved in recent years through the introduction of architecture description languages, methods, and tools to more fully address and characterize the quality attributes, allowing a richer (e.g., multi-dimensional) specification of constraints on how the defined functionality will be realized in the product, and facilitating additional analyses of the requirements and technical solutions.

  Some quality attributes will emerge as architecturally significant and thus drive the development of the product architecture. These quality attributes often reflect cross-cutting concerns that may not be allocatable to lower level elements of a solution. A clear understanding of the quality attributes and their importance based on mission or business needs is an essential input to the design process.

- Revised the subpractices in line with the above note.

CMMI Support for: understanding requirements - 9

In RD SP 3.4 Analyze Requirements to Achieve Balance

- Added the following new subpractice:

  4. Assess the impact of the architecturally significant quality attribute requirements on the product and product development costs and risks.

  When the impact of requirements on costs and risks seems to outweigh the perceived benefit, relevant stakeholders should be consulted to determine what changes may be needed.

CMMI Support for: understanding requirements - 10

In TS Introductory Notes

- Added technology maturation and obsolescence as additional drivers of requirements changes in maintenance and sustainment projects.

In VAL Introductory Notes

Reinforced when validation occurs in the product lifecycle.

“[snip] validation is performed early (concept/exploration phases) and incrementally throughout the product lifecycle (including transition to operations and sustainment).”

In VAL SP 1.1 Select Products for Validation

Added additional examples of products and product components that can be validated:

access protocols and data interchange reporting formats

Added example of validation method:

incremental delivery of working and potentially acceptable product

CMMI Support for: the architecture - 1

CMMI support for:

- creating/selecting
- documenting/communicating
- analyzing/evaluating

the architecture

Is mostly found in the first two goals of TS:

SG 1 Select Product Component Solutions

SP 1.1 Develop Alternative Solutions and Selection Criteria

SP 1.2 Select Product Component Solutions

SG 2 Develop the Design

SP 2.1 Design the Product or Product Component

SP 2.2 Establish a Technical Data Package

SP 2.3 Design Interfaces Using Criteria

SP 2.4 Perform Make, Buy, or Reuse Analyses

CMMI Support for: the architecture - 2

TS Informative Material Changes

"Quality attribute models, simulations, prototypes or pilots can be used to provide additional information about the properties of the potential design solutions to aid in the selection of solutions. Simulations can be particularly useful for projects developing systems-of-systems." *[TS Intro Notes]*

"Architectural ~~features~~choices and patterns that ~~provide a foundation for product improvement and evolution~~support achievement of quality attribute requirements are considered.
[snip] COTS alternatives [snip] can require modifications to aspects such as interfaces or a customization of some of the features to ~~better achieve product~~correct a mismatch with functional or quality attribute requirements, or with architectural designs." *[TS SG 1 note]*

CMMI Support for: the architecture - 3

TS Informative Material Changes (continued)

In TS SP 1.1 Develop Alternative Solutions and Selection Criteria

- Added an additional consideration for selection criteria:

  Achievement of key quality attribute requirements, such as product timeliness, safety, reliability, and maintainability

- Added new subpractice 4.

  4. Identify re-usable solution components or applicable architecture patterns.

In TS SP 2.1 Design the Product or Product Component

- Added additional examples of architecture definition tasks.

  –Selecting architectural patterns that support the functional and quality attribute requirements, and instantiating or composing those patterns to create the product architecture

  –Formally defining component behavior and interaction using an architecture description language

CMMI Support for: the architecture - 4

TS Informative Material Changes (continued)

In TS SP 2.2 Establish a Technical Data Package

- Added new subpractice 2.

  2. Determine the views to be used to document the architecture.

     Views are selected to document the structures inherent in the product and to address particular stakeholder concerns.

In TS SP 2.3 Design Interfaces Using Criteria

- Added to what "interface designs include:"
  - stimulus and data characteristics for software, including sequencing constraints or protocols
  - resources consumed processing a particular stimulus
  - Exception or error handling behavior for stimuli that are erroneous or out of specified limits.

CMMI Support for: implementing the system based on the architecture - 1

CMMI V1.3 support for implementing the system is mostly found in the third goal of the TS PA.

SG 3 Implement the Product Design

SP 3.1 Implement the Design

SP 3.2 Develop Product Support Documentation

TS Informative Material Changes

In TS SP 3.1 Implement the Design

- In Subpractice 1, added aspect oriented programming as a software coding methods example.

CMMI Support for: implementing the system based on the architecture - 2

Other Informative Material Changes

Special notes for Agile and for Product Lines have been inserted in the Intro Notes of various PAs in V1.3.

Changes Supporting Use of Agile Methods

Because CMMI practices are written for use in a broad variety of contexts, business situations, and application domains, it is not possible (even if it were appropriate) to advocate any specific implementation approach.

However, Agile methods and approaches are now in wider use, and so for V1.3, it seemed appropriate to acknowledge this, identify how Agile approaches can address CMMI practices and conversely, identify the value that CMMI can bring to Agile implementations.

The next set of slides describe how CMMI V1.3 addresses Agile methods.

Addressing Agile - 1

The Problem

Developers that use Agile methods sometimes resist using CMMI because they can't see how CMMI practices can complement or improve the effectiveness of Agile methods.

Overview of Solution

Added guidance to the appropriate PAs to do the following:

- Help users interpret the practices in a context where Agile methods are used
- Reinforce the applicability of the practices in an Agile environment
- Send the message that CMMI is a robust best practice framework meant to be used in Agile environments as well as other development environments

Addressing Agile - 2

Solution

Added a new section to DEV Chapter 5 entitled "Interpreting CMMI When Using Agile Approaches"

- This section describes how CMMI practices can apply in a variety of development environments. It also describes the interpretive guidance that has been added to selected PAs for use in Agile environments.

Added interpretive guidance to the following PAs:

- In DEV: CM, REQM, PP, RD, TS, PI, VER, PPQA, and RSKM
- In ACQ: AM, ATM, PMC, and PP
- In SVC: SSD

Added in DEV and SVC (SSD only) Agile-related examples as bullets in example boxes (informative material).

Addressing Agile - 3

A note added in the RD Intro Notes:

In Agile environments, requirements are communicated and tracked through mechanisms such as product backlogs, story cards, and screen mock-ups. [snip] Traceability and consistency across requirements and work products is addressed through the mechanisms already mentioned as well as during start-of-iteration or end-of-iteration activities such as "retrospectives" and "demo days." *[Emphasis added]*

A note added in the TS Intro Notes:

In Agile environments, the focus is on early solution exploration. By making the selection and tradeoff decisions more explicit, the Technical Solution process area helps improve the quality of those decisions, both individually and over time. [snip] When someone other than the team will be working on the product in the future, release information, maintenance logs, and other data are typically included with the installed product. To support future product updates, rationale (for trade-offs, interfaces, and purchased parts) is captured so that why the product exists can be better understood. [snip] *[Emphasis added]*

Addressing Agile - 4

For more information about using Agile in development and acquisition, and the relationship to CMMI, see:

- Glazer, Hillel; Dalton, Jeff; Anderson, David; Konrad, Mike; & Shrum, Sandy. *CMMI or Agile: Why Not Embrace Both!* (CMU/SEI-2008-TN-003). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, November 2008. http://www.sei.cmu.edu/library/abstracts/reports/08tn003.cfm
- Lapham, Mary Ann; Williams, Ray C.; Hammons, Charles; Burton, Daniel; and Schenker, Fred. *Considerations for Using Agile in DoD Acquisition* (CMU/SEI-2010-TR-022). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon® University, April 2010. http://www.sei.cmu.edu/library/abstracts/reports/10tn002.cfm
- McMahon, Paul E., "Integrating CMMI into Agile Development: Case Studies and Proven Techniques for Faster Performance Improvement." Addison-Wesley, 2011.

CMMI Support for: implementing the system based on the architecture - 3

Likewise, notes have been added to the Intro Notes of selected PAs to explain how the PA can be effectively applied in a product line environment.

Addressing Product Lines

An example of a note added in the RD Intro Notes:

For product lines, engineering processes (including requirements development) may be applied to at least two levels in the organization. At an organizational or product line level, a "commonality and variation analysis" is performed to help elicit, analyze, and establish core assets for use by projects within the product line. At the project level, these core assets are then used as per the product line production plan as part of the project's engineering activities. *[Emphasis added]*

An example of a note added in the TS Intro Notes:

For product lines, these practices apply to both core asset development (i.e., building for reuse) and product development (i.e., building with reuse). Core asset development additionally requires product line variation management (the selection and implementation of product line variation mechanisms) and product line production planning (the development of processes and other work products that define how products will be built to make best use of these core assets). *[Emphasis added]*

CMMI Support for: ensuring implementation conforms to the architecture - 1

CMMI support for ensuring the implementation conforms to the architecture is mostly found in the VER PA. (And also in notes and subpractices of PI SP 3.3 and TS SP 3.1 and 3.2.)

SG 1 Prepare for Verification
SP 1.1 Select Work Products for Verification
SP 1.2 Establish the Verification Environment
SP 1.3 Establish Verification Procedures and Criteria
SG 2 Perform Peer Reviews
SP 2.1 Prepare for Peer Reviews
SP 2.2 Conduct Peer Reviews
SP 2.3 Analyze Peer Review Data
SG 3 Verify Selected Work Products
SP 3.1 Perform Verification
SP 3.2 Analyze Verification Results

CMMI Support for: ensuring implementation conforms to the architecture - 2

In VER SG 1 Prepare for Verification

- Changed a note to read:

  Methods of verification include, but are not limited to, inspections, peer reviews, audits, walkthroughs, analyses, architecture evaluations, simulations, testing, and demonstrations.

In VER SP 1.1 Select Work Products for Verification

- Added additional examples of verification methods:

  software architecture conformance evaluation and continuous integration (i.e., Agile approach).

In VER SP 1.3 Establish Verification Procedures and Criteria

- Added new example of sources of verification criteria:

  customers reviewing work products collaboratively with developers

CMMI Support for: ensuring implementation conforms to the architecture - 3

In VER SP 2.1 Prepare for Peer Reviews

- In Subpractice 1, added additional example of types of peer review:

  architecture implementation conformance evaluation

In VER SP 2.3 Analyze Peer Review Data

- In Subpractice 4, added additional examples of peer review data that can be analyzed:

  user stories or case studies associated with a defect and

  the end-users and customers who are associated with defect

Software Engineering Institute | Carnegie Mellon CMMI V1.3 and Architecture
 123

CMMI Support for: evolving the architecture so that it continues to meet business and mission goals - 1

The need for evolution arises from both inside and outside:

"As the organization improves its process performance or as business strategies change, new business objectives are identified and associated quality and process performance objectives are derived." *[OPM SG 1 Notes]*

These objectives then drive the activities we read about in the project management and engineering PAs such as RD.

The phrase "establish and maintain" appears in the CMMI practices. It implies that key artifacts may need to change to remain useful (see next slide). If higher-level objectives change, the artifact may need to too.

As an example from RD:

"The modification of requirements due to approved requirement changes is covered by the "maintain" aspect of this specific practice; [snip]." *[SP 2.1 note]*

Software Engineering Institute | Carnegie Mellon CMMI V1.3 and Architecture
 124

CMMI Support for: evolving the architecture so that it continues to meet business and mission goals - 2

The definition for "establish and maintain" was changed in V1.3 to support the evolution described on the previous slide.

Establish and maintain

DEFINITION

Create, document, use, and revise . . . as necessary to ensure ~~it remains~~ they remain useful.

The phrase "establish and maintain" ~~means more than a combination of its component terms;~~ . . . plays a special role in communicating a deeper principle in CMMI: work products that have a central or key role in work group, project, and organizational performance should be given attention to ensure they are used and useful in that role.

This phrase has particular significance in CMMI because it often appears in goal and practice statements . . . and should be taken as shorthand for applying the principle to whatever work product is the object of the phrase.

Changes in CMMI Terminology - 1

Allocated requirement

Improved the definition and provided additional examples of what things requirements can be allocated to.

The improvements to the definition make the substance of the solution space and allocation of requirements to it more explicit, allowing for superior architectures and more insightful analyses (including verification) of requirements and technical solutions.

DEFINITION

Requirement that ~~levies~~ results from levying all or part of ~~the performance and functionality of~~ a higher level requirement on a lower level architectural element or design component.

More generally, requirements can be allocated to other logical or physical components including people, consumables, delivery increments, or the architecture as a whole, depending on what best enables the product or service to achieve the requirements.

Changes in CMMI Terminology - 2

Architecture

This term is included in the Glossary for the first time. (V1.2 used the phrase “product architecture” throughout but never defined it.)

This term and its use throughout the rest of the model is intended to encourage use of proven, architecture-centric practices and the recognition of “architecture” as a principal engineering artifact.

DEFINITION

The set of structures needed to reason about a product. These structures are comprised of elements, relations among them, and properties of both.

In a service context, the architecture is often applied to the service system.

Note that functionality is only one aspect of the product. Quality attributes, such as responsiveness, reliability, and security, are also important to reason about. Structures provide the means for highlighting different portions of the architecture. (See also “functional architecture.”)

Changes in CMMI Terminology - 3

Definition of required functionality and quality attributes

The “definition of required functionality” term has been removed from CMMI because of the implicit suggestion that functionality be addressed first or has highest priority. The term has been replaced with one that is intended to help ensure a sufficiently balanced focus (functional *and* non-functional) in requirements analysis.

DEFINITION

A characterization of required functionality and quality attributes obtained through “chunking,” organizing, annotating, structuring, or formalizing the requirements (functional and non-functional) to facilitate further refinement and reasoning about the requirements as well as (possibly, initial) solution exploration, definition, and evaluation.

As technical solution processes progress, this characterization can be further evolved into a description of the architecture versus simply helping scope and guide its development, depending on the engineering processes used; requirements specification and architectural languages used; and the tools and the environment used [snip].

Changes in CMMI Terminology - 4

“Functional analysis” and “functional architecture”

These terms are now “cul de sacs” in the model.

The only place these terms now appear in CMMI-DEV V1.3 outside the Glossary is in the first note of RD SP 3.2 and as an example work product.

The note contrasts the approaches implied by these terms with “modern engineering approaches” that encourage a more balanced treatment of requirements, functional and non-functional.

Changes in CMMI Terminology - 5

Product line

DEFINITION

A group of products sharing a common, managed set of features that satisfy specific needs of a selected market or mission~~.~~ and that are developed from a common set of core assets in a prescribed way.

The development or acquisition of products for the product line is based on exploiting commonality and bounding variation (i.e., restricting unnecessary product variation) across the group of products. The managed set of core assets (e.g., requirements, architectures, components, tools, testing artifacts, operating procedures, software) includes prescriptive guidance for their use in product development. Product line operations involve interlocking execution of the broad activities of core asset development, product development, and management.

Many people use “product line” just to mean the set of products produced by a particular business unit, whether they are built with shared assets or not. We call that collection a "portfolio," and reserve "product line" to have the technical meaning given here.

Changes in CMMI Terminology - 6

Quality attribute

This term is now included in the Glossary for the first time. The term is intended to supplant others – especially those focusing on only a few dimensions (e.g., "performance") – to encourage a broader view of non-functional requirements. The term was refined through much effort, as neither ISO 25030 (SQuaRE) nor the original SEI definitions were quite satisfactory.

DEFINITION

A property of a product or service by which its quality will be judged by relevant stakeholders. Quality attributes are characterizable by some appropriate measure.

Quality attributes are non-functional, such as timeliness, throughput, responsiveness, security, modifiability, reliability, and usability. They have a significant influence on the architecture.

Changes in CMMI Terminology - 7

Performance (not a term appearing by itself in Glossary)

One of our purposes for V1.3 was to achieve greater clarity in the engineering practices of CMMI. This purpose is aided when the term "performance," which has many meanings, is used unambiguously and correctly throughout. Thus, uses of the term "performance" were reviewed for clarity, and where appropriate, qualified, e.g.:

- supplier's performance
- project performance
- product performance
- technical performance
- organization's performance
- cost, schedule, performance
- performed process (CL1)
- process performance
- period of performance
- service delivery performance
- project progress and performance
- fit, form, function, performance

Related Changes

Product Integration

We revised PI SP 1.1 and the terminology used from an emphasis on "integration sequence" to an emphasis on "integration strategy" to reflect the complexity of product integration.

The product integration strategy describes the approach for receiving, assembling, and evaluating the product components that comprise the product.

SP 1.1 Establish an ~~Determine~~ Integration Strategy ~~Sequence~~

Establish and maintain a ~~Determine the~~ product ~~component~~ integration strategy ~~sequence.~~

Related changes were made elsewhere in the PI PA.

Presentation Outline

CMMI V1.3 – Context for modern engineering practices changes

Introduction to Architecture

Essential Architecture Practices

Where Are the Architecture-Centric Practices in CMMI V1.3?

Summary

Questions and Answers

Summary & Conclusions

The quality and longevity of a software-intensive system is largely determined by its architecture.

Early identification of architectural risks saves money and time.

There are proven practices to help ensure that suppliers and acquirers can develop and acquire systems that have appropriate architectures.

CMMI V1.3 has a new emphasis on architecture.

The efficacy of the architecture has a direct impact on program or mission success, and customer satisfaction.

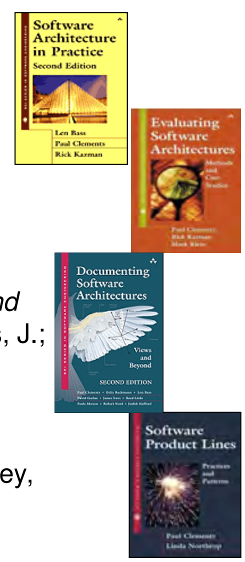
References - 1

Software Architecture in Practice, Second Edition
Bass, L.; Clements, P.; & Kazman, R. Reading, MA: Addison-Wesley, 2003.

Evaluating Software Architectures: Methods and Case Studies
Clements, P.; Kazman, R.; & Klein, M. Reading, MA: Addison- Wesley, 2002.

Documenting Software Architectures: Views and Beyond
Clements, P.; Bachmann, F.; Bass, L.; Garlan, D.; Ivers, J.; Little, R.; Nord, R.; & Stafford, J. Reading, MA: Addison-Wesley, 2002.

Software Product Lines: Practices and Patterns
Clements, P.; Northrop, L. Reading, MA: Addison-Wesley, 2001.



References - 2

You can find a moderated list of references on the "Software Architecture Essential Bookshelf"

http://www.sei.cmu.edu/architecture/start/publications/bookshelf.cfm

Grady Booch: Handbook of Software Architecture (currently only an on-line reference):

http://www.handbookofsoftwarearchitecture.com/index.jsp?page=Main

CMMI for Development, Version 1.3

http://www.sei.cmu.edu/library/abstracts/reports/10tr033.cfm

(also available as a book from the SEI Series on Software Engineering:)

Chrissis, Mary Beth; Konrad, Mike; & Shrum, Sandy. CMMI: Guidelines for Process Integration and Product Improvement, 3rd Edition. Boston: Addison-Wesley, 2011.

The SEI Software Architecture Curriculum

Three Certificate Programs

Six Courses	Software Architecture Professional	ATAM Evaluator	ATAM Leader
Software Architecture Principles and Practices*	✓	✓	✓
Documenting Software Architectures	✓		✓
Software Architecture Design and Analysis	✓		✓
Software Product Lines	✓		✓
ATAM Evaluator Training		✓	✓
ATAM Leader Training			✓
ATAM Observation			✓

✓: required to receive certificate

*: available through e-learning

Contact Information

Larry Jones
Research, Technology, and Systems Solutions Program
Telephone: 719-481-8672
Email: lgj@sei.cmu.edu

Mike Konrad
SEPM
Telephone: 412-268-5813
Email: mdk@sei.cmu.edu

U.S. Mail:
Software Engineering Institute
Carnegie Mellon University
4500 Fifth Avenue
Pittsburgh, PA 15213-3890

World Wide Web:
http://www.sei.cmu.edu/productlines
SEI Fax: 412-268-5758

Questions

SPI Manifesto
“Why You Need It”

CMMI Technology Conference 2010
Nov 2010
Denver, Colorado

Tim Kasse

Kasse Initiatives LLC
+1 – 972 - 987 - 7706 USA
+45 72 19 42 18 Europe

ようこそ

Welcome

WelKom

Huan Yín

Bienvenido

Bienvenue

Wilkommen

ΚΆΛΟΣ ΟΡΊΣΑΤΕ

Bienvenuto

Välkommen

Tervetuloa

Witamy

ברוכים הבאים

Pamelia Rost – EVP Business Development Kasse Initiatives

Discipline versus Agility

- Building quality software that has economic value has been, is, and will remain a “hard thing to do!”
- If one has strong discipline without agility, the result is classically bureaucracy and stagnation and possibly abandonment of process and planning altogether
- Claiming one is agile without discipline is the unbounded enthusiasm of a startup company that still has not made a profit and maybe never will
- The challenge is finding the right mix!

Agile Manifesto
for Software Development

Manifesto for Agile Software Development

- We are uncovering better ways of developing software by doing it and helping others do it.

- Through this work we have come to value:
  - **Individuals and interactions** over processes and tools
  - **Working software** over comprehensive documentation
  - **Customer collaboration** over contract negotiation
  - **Responding to change** over following a plan

Principles behind the Agile Manifesto

- Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
- Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- Business people and developers must work together daily throughout the project.

Principles behind the Agile Manifesto - 2

- Build projects around motivated individuals. Give them the environment and support they need, and <u>trust them to get the job done</u>.
- The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
- Working software is the primary measure of progress.
- Agile processes promote sustainable development. <u>*The sponsors, developers, and users should be able to maintain a constant pace indefinitely.*</u>

Principles behind the Agile Manifesto - 3

- Continuous attention to technical excellence and good design enhances agility.
- Simplicity--the art of maximizing the amount of work not done--is essential.
- The best architectures, requirements, and designs emerge from self-organizing teams.
- At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

Software Process Improvement Manifesto

The Inspiration for the SPI Manifesto

- **With models, standards, methods and techniques from all parts of the world focused on process and quality** it is only fitting that a process improvement manifesto was developed
- In September 2009, a group of 15 experts in Software Process Improvement (SPI) from all over the world gathered near Madrid, Spain and shared their expertise and wisdom from their many years of process improvement experience
- The meetings were held at the EuroSPI (European Software Process Improvement) conference (www.eurospi.net)
- Following the initial sharing, 30 workshop participants, Led by Jan Pries-Heje and Jorn Johansen, brainstormed core values and principles specifically focused on process improvement

Chief Editors
Jan-Pries-Heje - Roskilde University
Jorn Johansen – Delta Axiom

Process

- Process defines how a business does business and may include a set of processes such as:
  - Software Engineering processes
  - Hardware Engineering processes
  - Systems Engineering processes
  - Manufacturing processes
  - Financial processes
  - Human Resources processes
  - Legal processes
  - ………..

Process - 2

- Process helps to establish the business culture and then sets guidelines and expectations
- Process can be viewed as a methodology that is applied from elicitation of requirements to design through delivery
- There are no shortcuts – there are no other alternative methods that a business can adopt that embraces a **“cradle to grave”** philosophy to ensure quality and profitability with ***control*** every step of the way

Process - 3

- We build the business right – through process
- We build the right business – with guarantees of product and service quality and customer satisfaction
- Process is the fastest-lowest cost path to get there and know if you are there!

SPI Manifesto

http://www.madebydelta.com/imported/images/DELTA_Web/documents/Ax/SPI_Manifesto_A.1.2.2010.pdf

Values and Principles

Values

- A Value is something that deserves to be held up because of its importance or worth
- The SPI Manifesto prioritized values of people, business focus, and a belief that organizational change is at the core of Software Process Improvement

Values Overview

- Values
  - People – Must involve people actively and affect their daily lives not to be focused on management alone
  - Business – What you do to make business successful – this is not about living to deploy a standard, reach a maturity level, or obtain a certificate even though it can certainly help do all of those things
  - Change – Process improvement is inherently linked with change – we realize and accept that we cannot continue to live as we do today – we must change – perhaps a little or perhaps a lot

Values Details

People

- ***We truly believe that SPI must involve people actively and affect their daily activities***
- Context and Problem
  - The last decade has brought "Ivory Towers" using magic tools and models that paint process diagrams
  - In most organizations, the projects and service providers did not really use their organizational processes
  - The people who were most affected were not involved in the process description development

People - 2

- Value Explained
  - Business success depends on the competitiveness of the organization
  - The competitiveness of every organization is based on the knowledge, engagement, and commitment of the people working in it
  - Only active involvement of the people working in the organization ensures the success of a SPI initiative from the business perspective
  - Actively involved people need sufficient information and training on how to operate on that information

People - 3

- Hints and Examples
  - The modern organization paradigm is having its people solving problems and changing the organization together
    - Having experts solve the problems and forcing change on the rest of the organization's workforce has not and does not work
  - Enablers for success in modern organizations include:
    - People making full use of their experience
    - People taking responsibility for change on their project and throughout their organization
    - People using and improving the processes they have helped to define

Business

- ***We truly believe that SPI is what you do to make business successful***
- Context and Problem
  - Many people do not believe that they need processes in order to build and deliver software products
  - Process is too often seen as somebody else's process description and not applicable
  - Processes are often forces on projects that do not fit the need of the project or the business

Business - 2

- Value Explained
  - Process descriptions are just words – We believe that the process should bring value to the business
  - For successful process improvement we must ensure that any improvement recommendations are targeted to the actual business-related objectives
    - Not just try to be compliant with a standard or model
  - Process should reflect how the work actually gets done – it should not be a set of words that projects must ignore to be successful
    - Words and actions need to be consistent
    - "We get the job done in spite of the processes and management"

Business - 3

- Hints and Examples
  - Use today's project / organizational implemented processes as an agreed upon baseline for process improvements
  - Understand the vision and business objectives to ensure the process can always be shown to support them
  - Always refer to the process description as a representation of the process
  - Communicate how standards and models are meant to support process improvement
  - Practice continuous communication at all levels of management and practitioners

Change

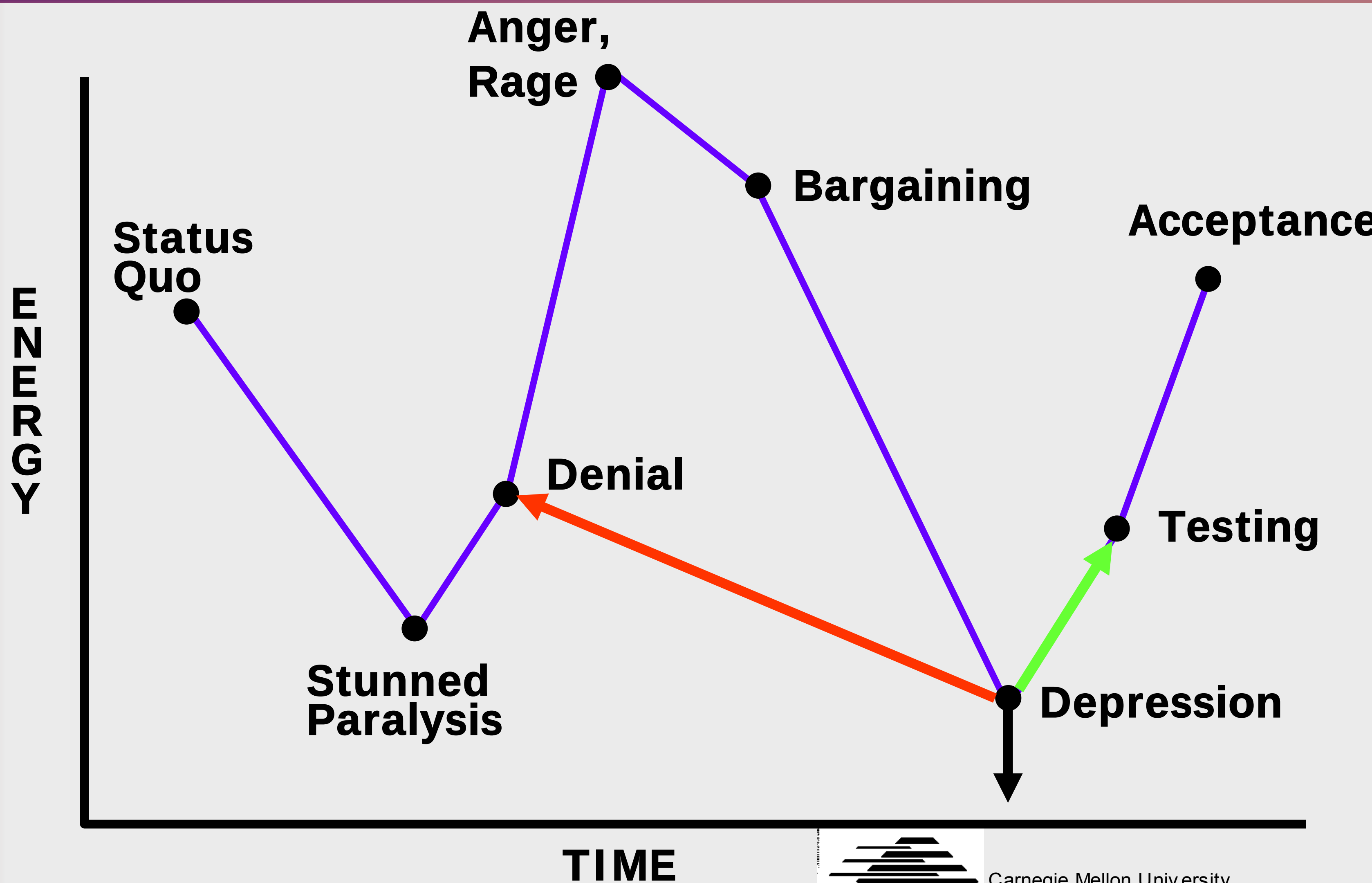
- ***We truly believe that SPI is inherently linked with change***
- Context and Problem
  - Improvement involves change for the individual, the project, and the organization
    - Maybe the change is small or maybe it is extensive but there will be change and many managers and developers do not want change in their environment and especially in themselves
    - We know that it is difficult for people to accept or adopt change, because they are comfortable doing things they way they always have even if it costs them overtime or loss of social interaction

Managing Complex Change Requirements

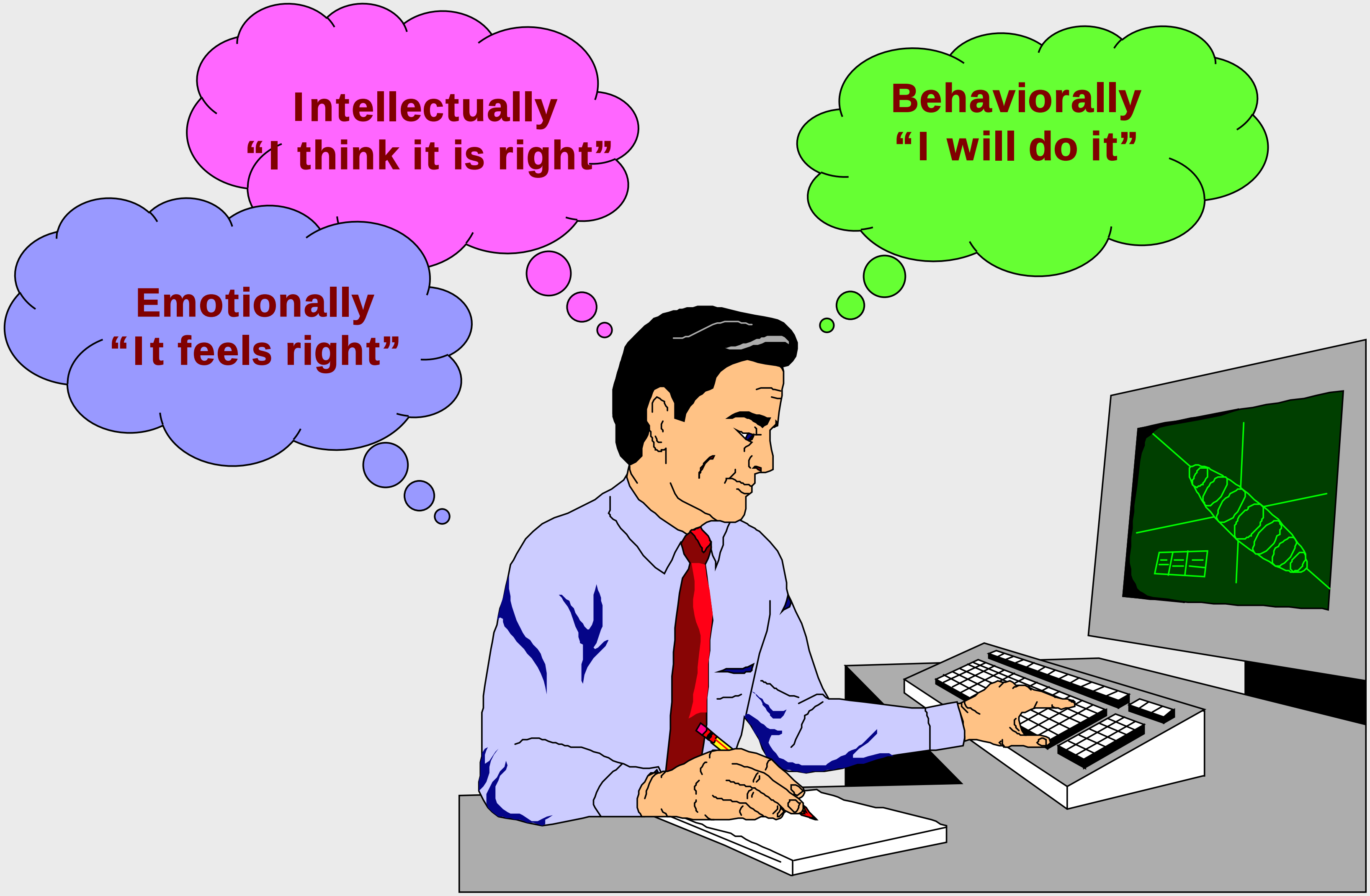
VISION	SKILLS	INCENTIVES	RESOURCES	ACTION PLAN	CHANGE
	SKILLS	INCENTIVES	RESOURCES	ACTION PLAN	CONFUSION
VISION		INCENTIVES	RESOURCES	ACTION PLAN	ANXIETY
VISION	SKILLS		RESOURCES	ACTION PLAN	SLOW CHANGE
VISION	SKILLS	INCENTIVES		ACTION PLAN	FRUS-TRATION
VISION	SKILLS	INCENTIVES	RESOURCES		FALSE STARTS

"Managing Technological Change"
Carnegie Mellon University
Software Engineering Institute

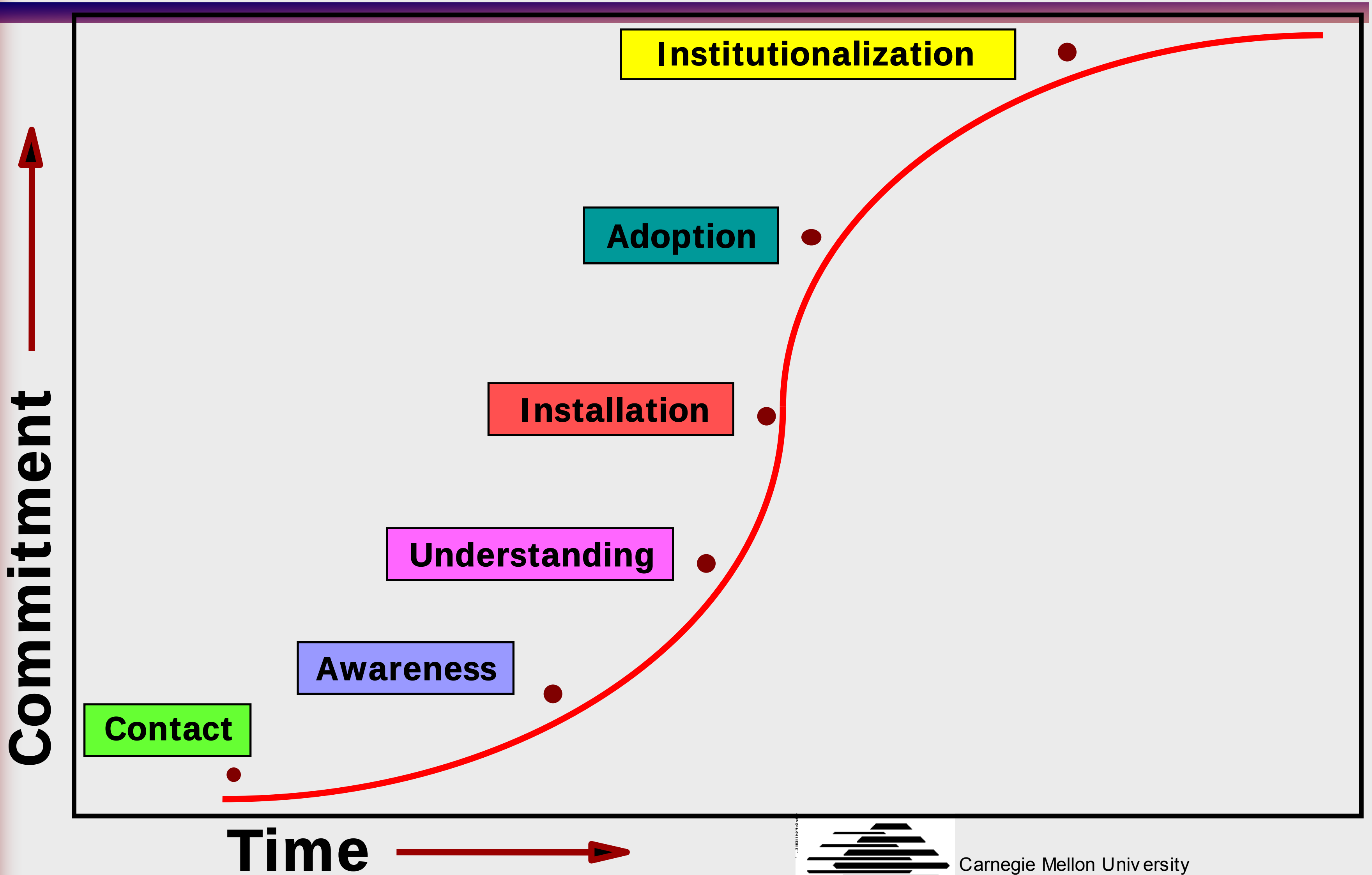
The Response to Change



Three Ways People Respond To Change



Commitment is a Phased Process



Change - 2

- Value Explained
  - If we accept that process improvement means change, then our process improvement initiative must have a change management component in it
  - Process improvement is important for product quality, customer satisfaction and measurable business but we want it together with satisfied employees

Change - 3

- Example
  - IT organization in a predominantly Asian culture started a process improvement initiative
  - One change required was to institutionalize Peer Reviews
  - However, colleagues did not want to review their peers work and find major defects for fear of causing them to lose face
  - Training, retraining, videotaping, and coaching did not produce the desired results from Peer Reviews after 3 years

Change - 4

- Consultant explained that if the major defects were not found in Peer Reviews they would be found by the customer and everyone would lose face including the CEO
- CEO appointed middle managers to serve as coaches and encouraged the project members to fully participate in the Peer Reviews as they were intended to function
- Management's commitment to change encouraged the practitioners to participate in the Peer Reviews
- Result: No one got fired | product quality went up | jobs were kept | profits increased | and lifestyles were improved due to less time needed in finding defects
- CEO declared that this culture change was the most significant event in the process improvement initiative!

Principles Details

Principles

- A Principle is something that can serve as a foundation for action!
- The ten (10) principles developed to support the SPI Manifesto values are intended to be used to govern personal behavior in relation to Software Process Improvement work

Principles Overview

- People
  - **Principle 1** – Know the culture and focus on needs
  - **Principle 2** - Motivate all people involved
  - **Principle 3** - Base improvement on experience and measurements
  - **Principle 4** - Create a learning organization

Principles Overview - 2

- Business
  - **Principle 5** - Support the organization's vision and business objectives
  - **Principle 6** - Use dynamic and adaptable models as needed
  - **Principle 7** - Apply risk management

Principles Overview - 2

- Change
  - **Principle 8** - Manage the organizational change in your improvement effort
  - **Principle 9** - Ensure all parties understand and agree on process
  - **Principle 10** - Do not lose focus!

People Principles

Principle 1 - Know the Culture and Focus on Needs

- Explanation
  - The culture of an organization is fundamentally embedded in human behavior
    - It is expressed through norms (explicit or implicit) that the organization used to express behavioral expectations
    - Culture also provides an indication of appropriate and inappropriate attitudes and behaviors
    - These rules also affect the interactions with others
  - The organizational culture is a shared system of meanings, values, and practices by the employees in the organization

Principle 1 - Know the Culture and Focus on Needs - 2

- Practices are distinguishable characteristics of the organizational culture that have a deep meaning for the members of the organization but are usually invisible to outsiders at a glance
- Values are "qualities,", principles, and behaviors considered to be morally or intrinsically noble, valuable and desirable by the members of the organization
- Cultural values are deeply ingrained and are held closely even if conflict results

Principle 2 - Motivate All People Involved

- Explanation
  - Process improvement does not succeed by defining processes in a “highly sophisticated” process group
  - Use the experience of the functional experts to define and improve those parts of the process that affect them in their daily work
    - Empowered experts will bring the necessary skills and the right mix of competence in order to achieve real value
  - Management support, promoted by Deming is always imperative to have
  - People need to be allowed to ask, “What is in it for me?”
    - Overt resistance is better than covert resistance!

Principle 2 - Motivate All People Involved - 2

- Coordination and cooperation between all levels of management and practitioners will ensure a widely accepted process and commitment of all of the people

- We recommend providing the necessary resources like training, equipment, and coaching support to all people who are expected to use their project's and/or organization's processes

- We also recommend reviewing the organization's reward structure and modifying it appropriately to support projects who follow processes with business success and not just put "heroes" in the spotlight

YOUR EXAMPLES

Principle 3 - Base Improvements on Experience and Measurement

- Explanation
  - As processes are developed from what people do, any process improvement effort must seek to optimize this "doing"
  - Conditions for optimization can be discussed but only the individual can change his/her actions
    - This requires individual competencies, readiness, and willingness to learn and optimize actions

Principle 3 - Base Improvements on Experience and Measurement - 2

- Readiness is obtained through experience as well as input or visible measurements of process capabilities
- Competence sets your ability to reflect on your actions based on experience, input, and measurements
  - This new knowledge will help change future actions
- Willingness motivates you to step through the learning cycle
  - It is influenced by the organization's culture, your own personality, incentives, requests or orders

Principle 4 - Create a Learning Organization

- Explanation
  - A practice accepted by all levels of managers and practitioners that represents useful core knowledge in a learning organization has the following three distinctive features:
    - For developers it has practical value to improve the existing development work
    - For managers it helps to save time, cost, and to increase quality
    - For assessors it helps to demonstrate improved capability
  - Such practices are disseminated across all projects in the learning organization

Principle 4 - Create a Learning Organization - 2

- We highly recommend that you work toward turning your organization into a "learning organization" that continuously facilitates the learning of its members and shares practical process experience across projects

YOUR EXAMPLES

Business Principles

Principle 5 - Support the Organization's Vision and Business Objectives

Explanation

- Dr. W. Edwards Deming stated in most of his books and lectures: "Process improvement should be done to help the business – not for its own sake."
- Process improvement initiatives should, as a minimum, be able to demonstrate the following:
  - Traceability to the organization's vision statement
  - Clearly stated business objectives that support the vision and are able to guide the organization's and project's efforts to produce measurable results
  - Measurement and analysis objectives that are aligned with established "information needs" and business objectives
  - Objective results that can be used in making business judgments and taking appropriate corrective actions

Principle 6 - Use Dynamic and Adaptable Models as Needed

- Explanation
  - Models do not depict the real world but represent a simplified view of the real world
  - Process improvement in general, is not tied to any model but is tied to the organization's business objectives and needs
    - Models include CMMI and SPICE
    - Standards include ISO 9001 and ISO 9126
    - Techniques / approaches include Six Sigma, SCRUM and Agile
    - Lifecycle models include Waterfall, Incremental, V-Model, Spiral, and Evolutionary

Principle 6 - Use Dynamic and Adaptable Models as Needed - 2

- Experience has shown that in most cases, you cannot simply follow one model or standard and expect to get the best results
  - Models and the concepts built into them can and should be combined to achieve business objectives
- Each model and standard should be thought of as a tool box that can help to resolve specific organizational challenges
- The best models have a dynamic component to them
  - They have built-in ways to take circumstances and contingencies into account

Principle 7 - Apply Risk Management

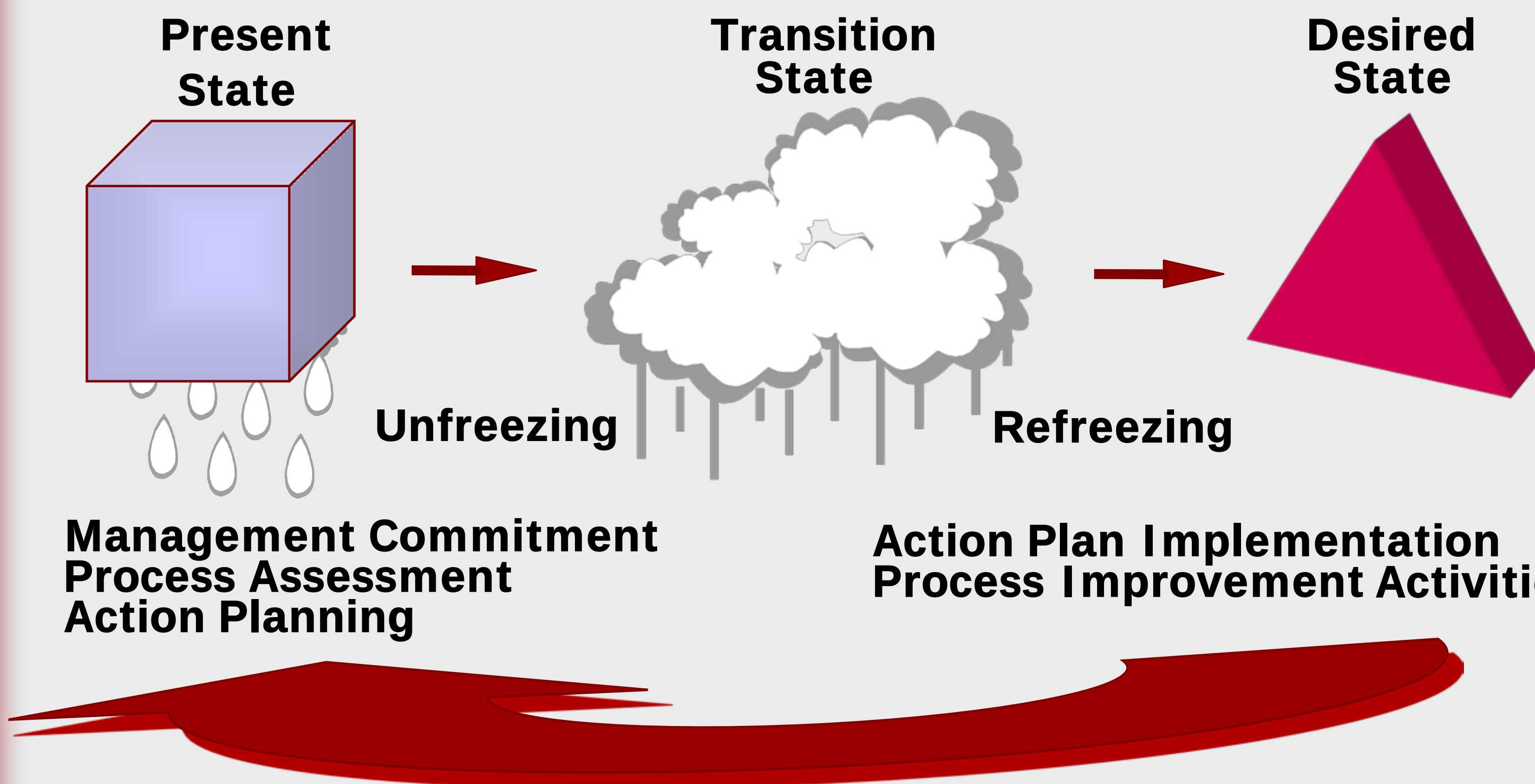
- Explanation
  - Any improvement effort may go wrong or not work as expected
    - This does not mean the process improvement initiative or the model or standard chosen to support it is wrong
  - Project management standards such as the one developed by the Project Management Institute has risk management built in as an integral part of any successful project planning
  - If you view the process improvement initiative as a project, which we recommend, you should consider what might go wrong before processes are developed and placed into the projects and developers hands

Change Principles

Principle 8 - Manage the Organizational Change in Your Improvement Effort

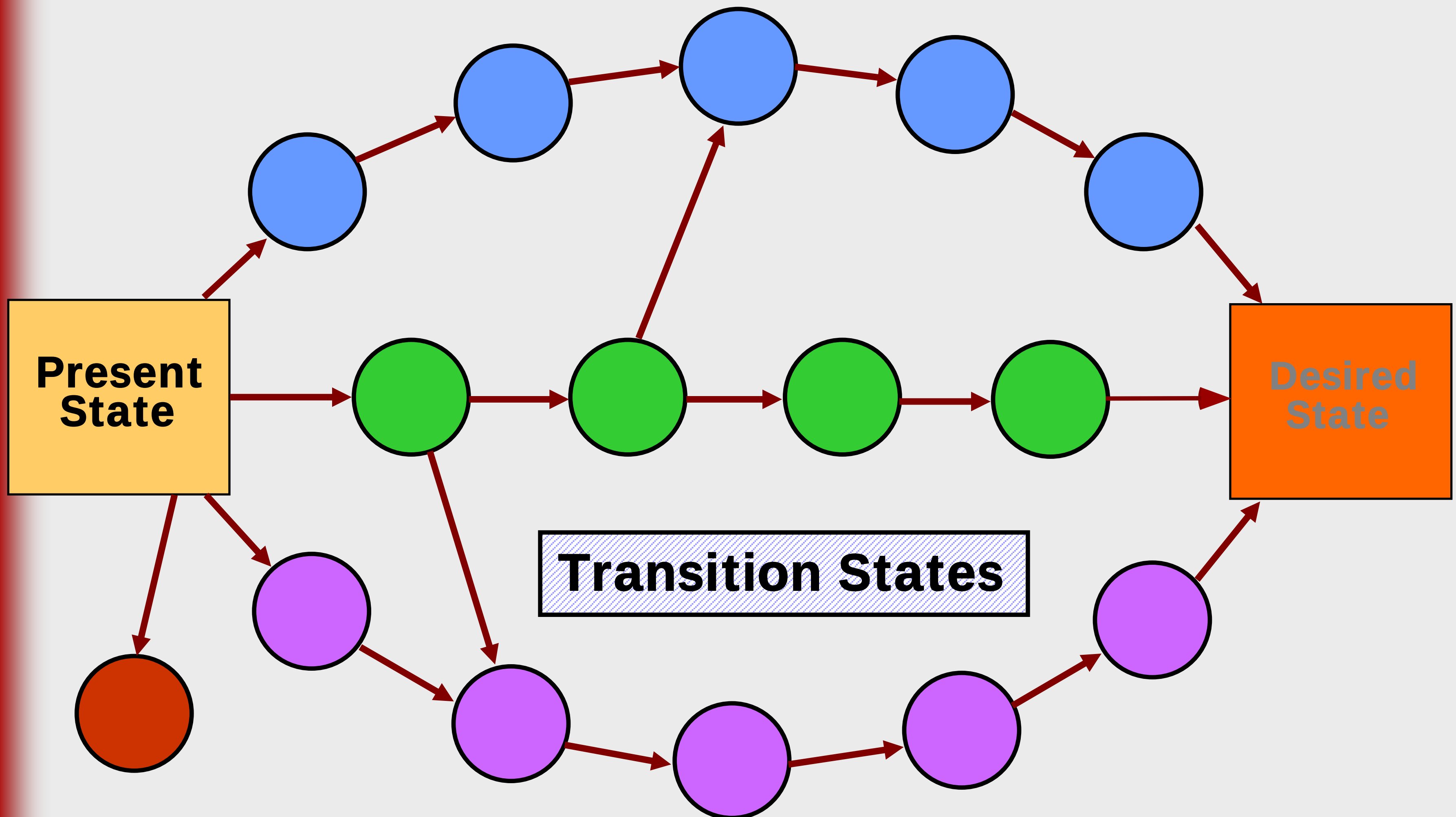
- Explanation
  - Real, measurable improvement requires real people to really change their behavior!
    - Process improvement is about organizational change
  - The simplest depiction of organizational change is the three-step model: Unfreeze – Move (Transition) – Freeze as shown in the following slides

A Simple Change Model



Carnegie Mellon University
Software Engineering Institute

A Sample Change Model



Principle 8 - Manage the Organizational Change in Your Improvement Effort - 2

- Unfreeze – to unfreeze for process improvement, you have to make the organization “receptive” to change
  - The organization must realize there is a need for change
  - There should be relevance to the individuals in the workforce
  - Unfreezing is needed because if you do not recognize the need for this step, and create organizational receptivity, the organization will behave like a block of ice, it will naturally resist change

Principle 8 - Manage the Organizational Change in Your Improvement Effort - 3

- **Move / Transition** – to move your improvement effort, a solution to the relevant problem that was identified during the unfreezing process should be proposed
  - Project Managers and Members need to be able to count on and receive coaching and in-the-trenches support
  - Don't forget the "bathtub effect" – When a new process or tool or technique is introduced into a project, the productivity of the project members will get worse before it gets better
    - Coaching in the trenches where the practitioners live can help reduce the dip in productivity when the process is introduced

Principle 8 - Manage the Organizational Change in Your Improvement Effort - 4

- Freeze – make sure the change is a permanent part of how the organization works
  - Policies – describing the required behavior change
  - Training, mentoring, coaching
  - Tool support
  - Management “walking the talk”
  - Measurements and feedback so the managers and practitioners see and continue to see the benefits of the change

YOUR EXAMPLES

Principle 9 - Ensure All Parties Understand and Agree on Process

- Explanation
  - Process descriptions are a snapshot of some important part of the organizational common agreement on how the organization works
    - But the process description are only valuable if they are agreed upon by the workforce
  - Process descriptions can and often are packaged into models and standards such as CMMI, SPICE, and ISO 9001
  - Process improvements constantly challenge the models and process descriptions but this is a “good thing”
    - Processes that are continuously reviewed and improved as the organization’s business and constraints change will remain practical and used
    - If they are allowed to remain stagnant the process improvement may grind to a halt or to back to being only project or individually owned

Principle 9 - Ensure All Parties Understand and Agree on Process - 2

- To ensure "living" operational and adaptive models and processes the organization must ensure they are:
  - Flexible and tailorable – usable for different types of projects in the organization
  - Expressed in a common language and visualized when possible
  - Based on communicated, understood, commonly agreed upon, and supported process improvement proposals
    - They are developed, deployed, and continuously maintained

Principle 10 - Do Not Lose Focus

- Explanation
  - Define targets, plan the measures to reach the targets, and stick to the improvement plan
  - Each improvement has to make a contribution to better fulfill the business goals and offer people motivation for changing their behavior
    - Without business impact, it is not possible to get a budget for measures
    - Without involvement of the people, the measures will not lead to a change of behavior
  - Appropriate measures have to be agreed on with relevant stakeholders at all hierarchical levels

Principle 10 - Do Not Lose Focus - 2

- Integrate process improvement actions into daily operational activities and carry them out with the same persistence as any other aspect of the daily business
- Provide for continuous motivation of the workforce to avoid the risk of the process improvement effort becoming uninteresting or boring

Principle 10 - Do Not Lose Focus - 3

- Companies which are consequent in SPI and do not lose focus will see many benefits including:
  - Increased efficiency
  - Better product quality through better processes
  - Trust from customers because of demonstrable high capability levels
  - Competitive advantage for new business
  - Employees who are willing to participate in SPI on an ongoing basis – true continuous process improvement!

YOUR EXAMPLES

Summary

- Now it is time to use the SPI Manifesto!
- Jorn Johansen and Jan-Pries-Heje, the leaders and chief editors of the SPI Manifesto put forth a reminder on what to use the manifest for.
  - You can use the manifest to obtain knowledge of SPI.
    - It will help you remember what is important about software process improvement
    - Each value and the consequent principles are written so you can easily place yourself into the problem and context
    - Short explanations for each value are provided that can further augment your understanding
    - Each value also has some relevant examples that will make it easier to learn and remember the values and principles

Summary - 2

- You can use the SPI Manifesto when you are responsible for planning a SPI project
- You can apply these SPI Manifesto principles in your organization's process improvement project that will support the necessary corresponding change

- Thanks is given to all that shared their experience and worked together to produce this SPI Manifesto but we have not stopped………
  - The next three years at the EuroSPI conferences, additional workshops will be established to substantiate the values and related principles and to "live" continuous process improvement through improvement to the SPI Manifesto

SPI Manifesto Workshop Participants

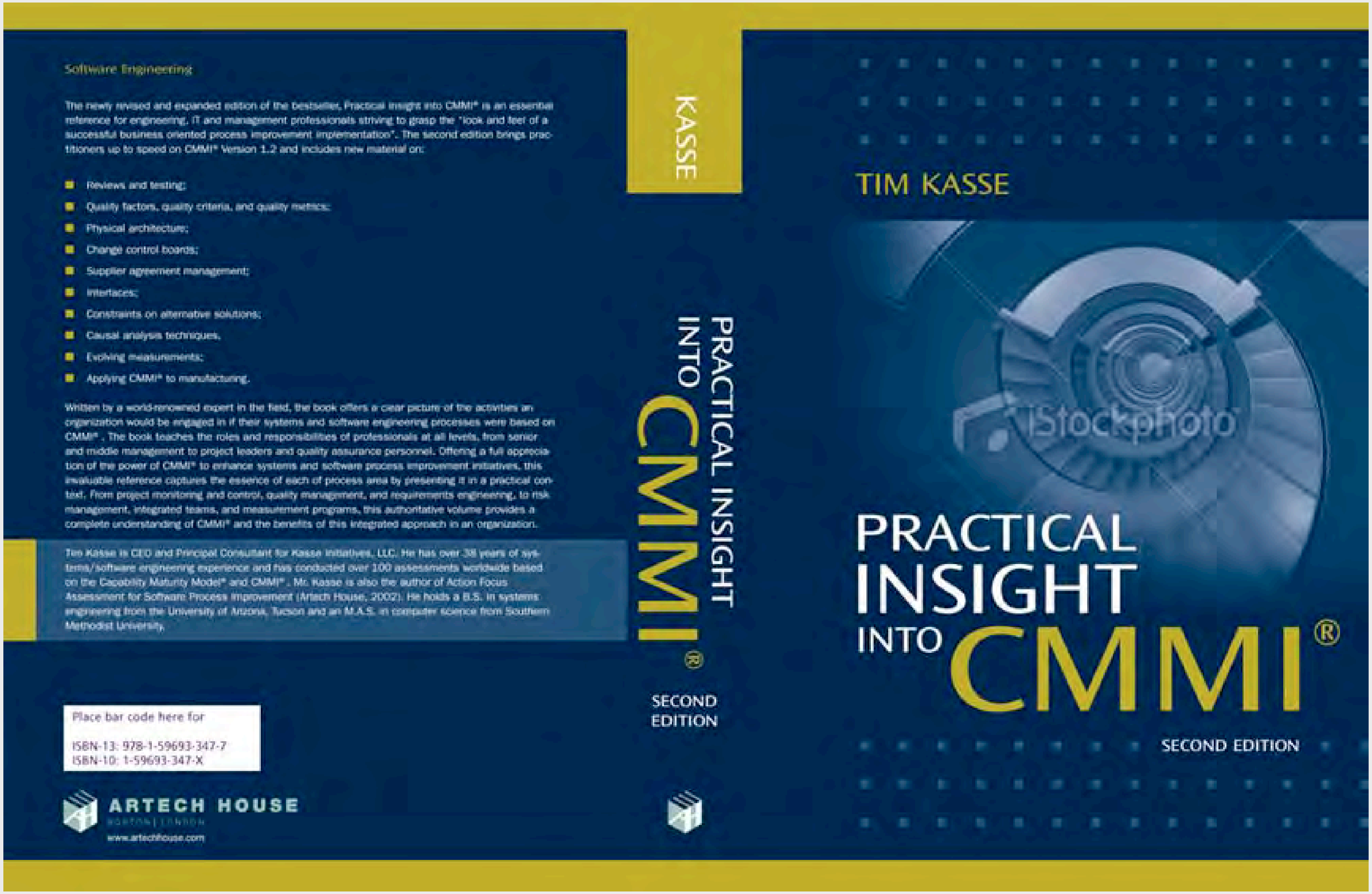
The SPI Manifesto Contributors

- Bo Balstrup
- Miklos Biro
- Alec Dorling
- Kurt S. Frederichsen
- Keith Fuller
- Sema Gazel
- Elli Georgiadou
- Christian Hertneck
- Jørn Johansen
- Tim Kasse
- Diana Kirk
- Kouichi Kishida
- Morten Korsaa
- Patricia McQuaid
- Richard Messnarz
- Risto Nevalainen
- Rory O´Connor
- Tom Peisl
- Jan Pries-Heje
- Andreas Riel
- Clenio F. Salviano
- Jürgen Schmied
- Tomas Schweigert
- Bernhard Sechser
- Kerstin Siakas
- Gunther Spork
- Serge Tiechkiewitch
- Detlef Vohwinkel
- Thomas Wegner

Tim Kasse

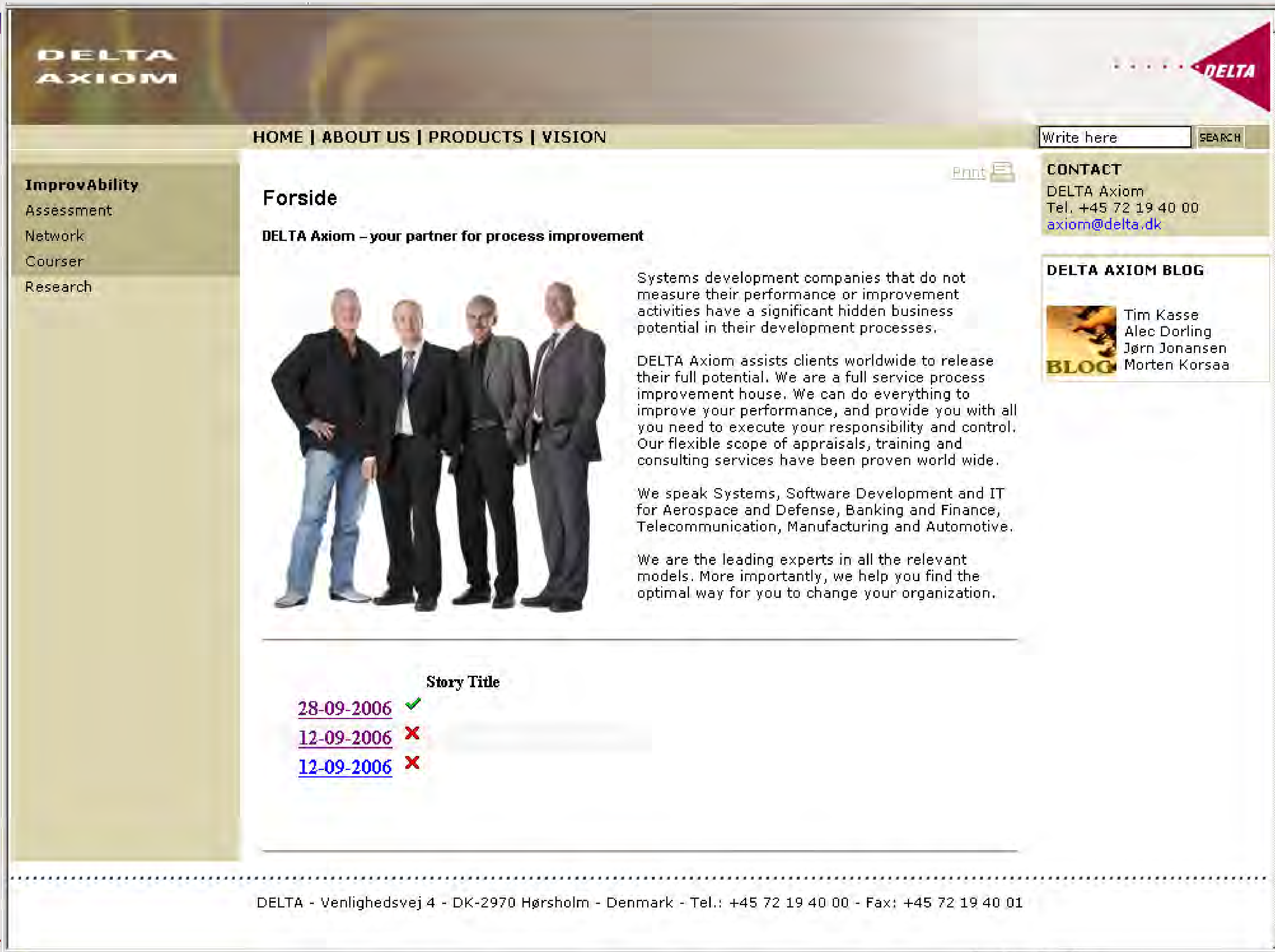
- CEO and Principal Consultant of Kasse Initiatives
- Visiting Scientist - Software Engineering Institute
- Visiting Fellow - Institute for Systems Science / National University of Singapore
- Author of Action Focused Assessment for Software Process Improvement
- Author of Practical Insight Into CMMI

Books From Kasse Initiatives



Books
From Kasse Initiatives - 2

Delta Axiom


DELTA AXIOM
DELTA
HOME | ABOUT US | PRODUCTS | VISION
Write here
SEARCH
ImprovAbility
Assessment
Network
Courser
Research
Print
Forside
DELTA Axiom – your partner for process improvement
Systems development companies that do not measure their performance or improvement activities have a significant hidden business potential in their development processes.
DELTA Axiom assists clients worldwide to release their full potential. We are a full service process improvement house. We can do everything to improve your performance, and provide you with all you need to execute your responsibility and control. Our flexible scope of appraisals, training and consulting services have been proven world wide.
We speak Systems, Software Development and IT for Aerospace and Defense, Banking and Finance, Telecommunication, Manufacturing and Automotive.
We are the leading experts in all the relevant models. More importantly, we help you find the optimal way for you to change your organization.
CONTACT
DELTA Axiom
Tel. +45 72 19 40 00
axiom@delta.dk
DELTA AXIOM BLOG
BLOG
Tim Kasse
Alec Dorling
Jørn Jonansen
Morten Korsaa
Story Title
28-09-2006
12-09-2006
12-09-2006
DELTA - Venlighedsvej 4 - DK-2970 Hørsholm - Denmark - Tel.: +45 72 19 40 00 - Fax: +45 72 19 40 01

Method Park

Bernd Hindel
CEO & Founder of Method Park

Tim Kasse – CEO & Principal Consultant
Kasse Initiatives LLC

Effective Technology Transition Strategies

CMMI Technology Conference 2010
Nov 2010
Denver, Colorado

Tim Kasse

Kasse Initiatives LLC
+1 – 972 - 987 - 7706 USA
+45 72 19 42 18 Europe

Welcome

ようこそ

WelKom

Huan Yín

Bienvenido

Bienvenue

Wilkommen

ΚΆΛΟΣ ΟΡΊΣΑΤΕ

Bienvenuto

Välkommen

Tervetuloa

Witamy

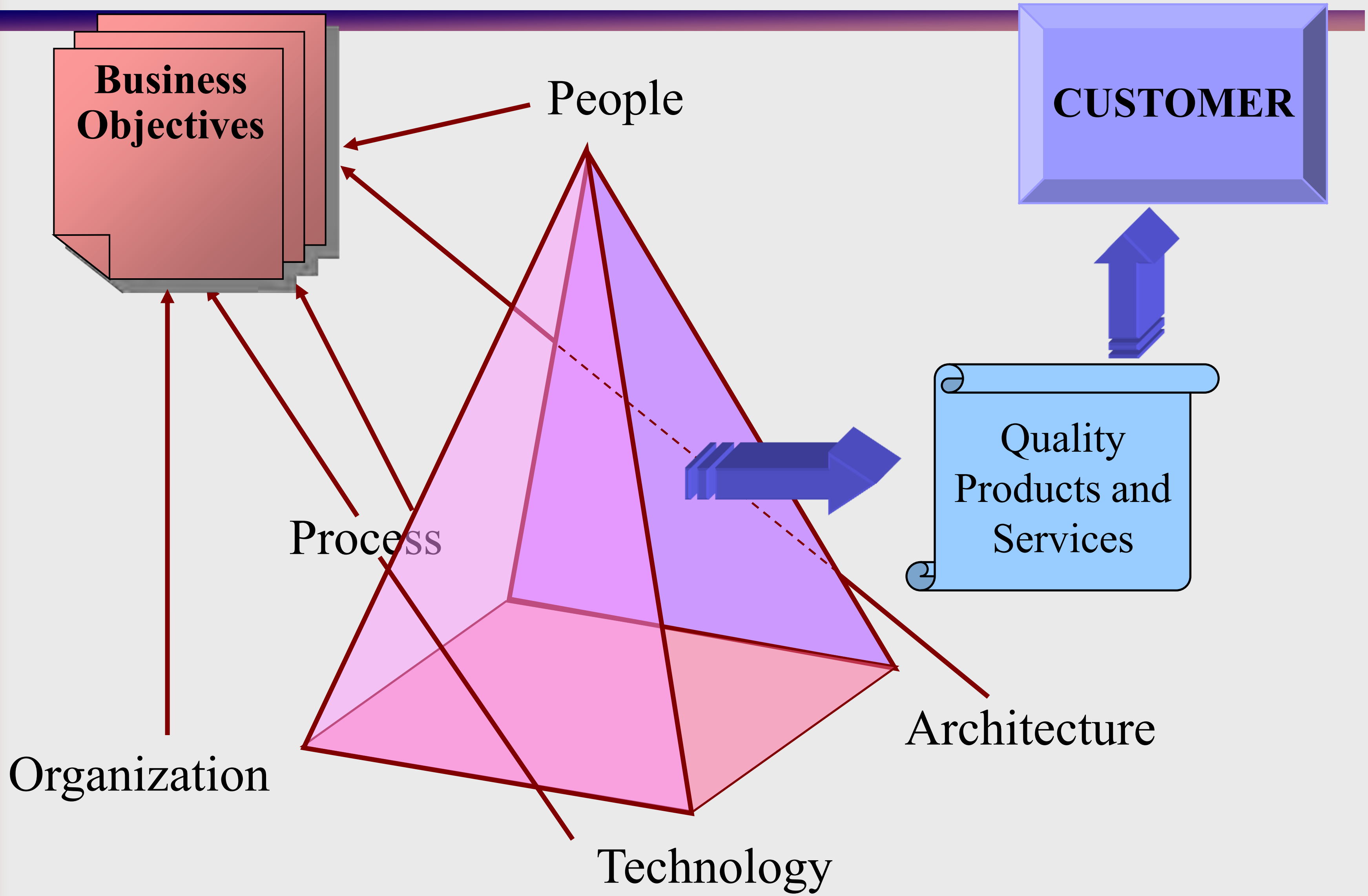
ברוכים הבאים

Pamelia Rost – EVP Business Development Kasse Initiatives

General Definitions of Process

- Process – a sequence of steps performed for a given purpose (IEEE)
- Process – the logical organization of people, materials, energy, equipment, and procedures into work activities designed to produce a specified end result (From Pall, Gabriel A. Quality Process Management. Englewood Cliffs, N.J.: Prentice Hall, 1987.)

Business Process Perspective



Process Improvement for Business

- Process improvement should be done to help the business—not for its own sake.

“In God we trust,
all others bring data.”
- W. Edwards Deming

Supporting Senior Management's Vision

Vision

- The purpose of the visionary questions is to make sure that the improvement program is aligned with senior management's vision
  - Where does senior management think the organization will be in the next year, and in the next two to five years?
  - What products will be in the mainstream?
  - Who will the competitors be?
  - Where will the collaborators or strategic alliance partners come from?
    - From what industry will they come from?
  - What technology changes are expected and/or will be required to support the vision?

Vision - 2

- What does the organizational structure have to be to support this vision?
- Who will the organization‘s suppliers be?
- What kind of organizational culture would you like to have to support this vision?
- What are the quality goals that are expected to be realized?
- How will a Process Improvement Initiative based on the CMMI and other related models and standards support this vision?
- What skills will your workforce need to support the vision?
- What skills will you as the Senior Management Team need to support the vision?

Supporting the Organization's Business Objectives

Business Objectives

- For a focus on Process Improvement to be successful, it must be tied to the organization's business objectives:
  - What are the organization's highest priorities?
  - What business consequences have resulted from weak or ineffective focus on quality management functions?
  - What action is being taken to correct the cause?
  - How can a focus on Process Improvement support the organization's business objectives

Business Objectives - 2

- Examples of Business Objectives
  - Reduce time to market
  - Reduce system errors that are discovered by customers
  - Improve delivery time
  - Increase quality of products
  - Find and fix software defects once and only once
  - Reduce project risks
  - Gain control of suppliers
  - Improve service delivery
  - Improve service availability and capacity
  - Shorten find to fix repair rate

Supporting the Organization's Measurement Objectives

Measurement Objectives

- While establishing measurement objectives, a project/organization should:
  - Document the purposes for which measurement and analysis is done
    - What is the information needed?
    - Are measures available to satisfy the information needed?
    - Is the frequency of the collection of the base measure high enough?
  - Specify the kinds of actions that may be taken based on the results of the data analyses
  - Ensure business objectives and measurement objectives are developed with clear —WHYs” this measure will support the business and quality goals of the project and organization

Helping Project Leaders to Manage and Control Better

Process Improvement: What Value to Project Leaders?

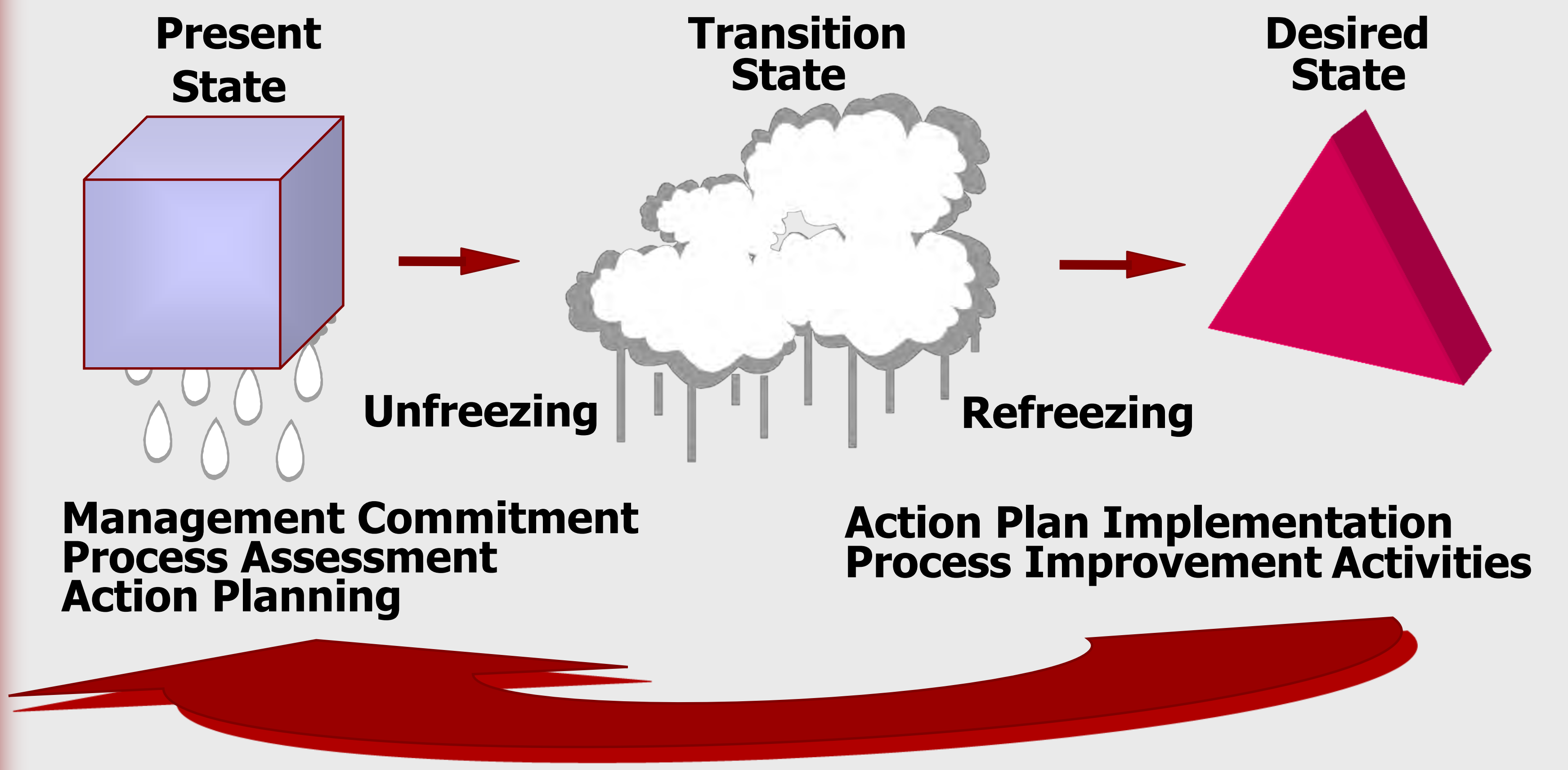
- What measurable value will the quality management initiative bring to the project leaders who bear the line responsibility for product delivery?
  - More accurate schedules?
  - Higher productivity of developers?
  - Better quality products?
  - Traceable requirements?
  - Controlled configuration items?
  - Reviews focused on critical components?
  - Better control of suppliers?
  - Reduction in potential risks?

Process Improvement Means Change

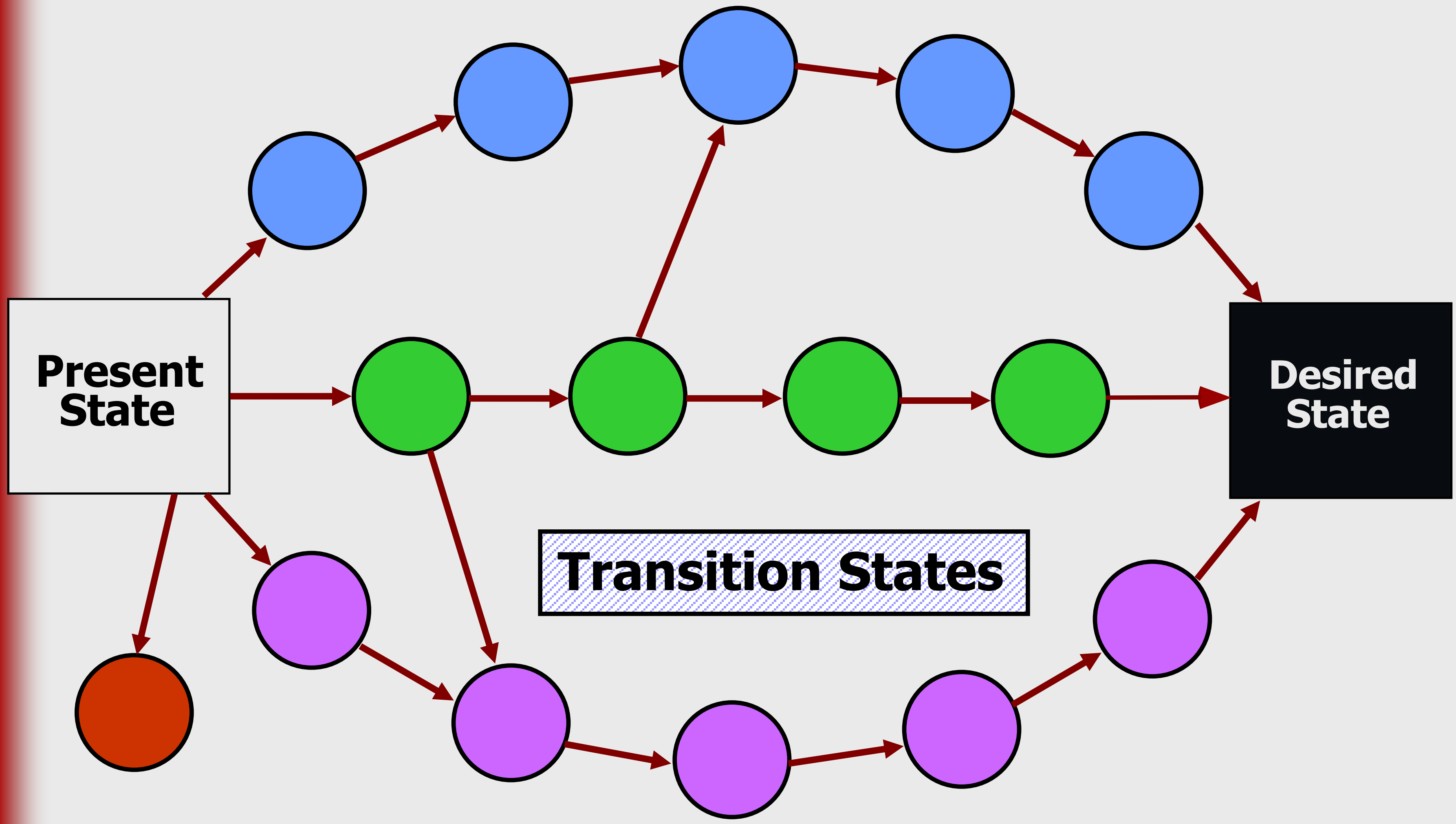
Principles of Process Change

- Major changes must be sponsored by Senior Management
- Focus on fixing the process, not assigning the blame
- Understand current process first
- Change is continuous
- Improvement requires investment
- Retaining improvement requires periodic reinforcement

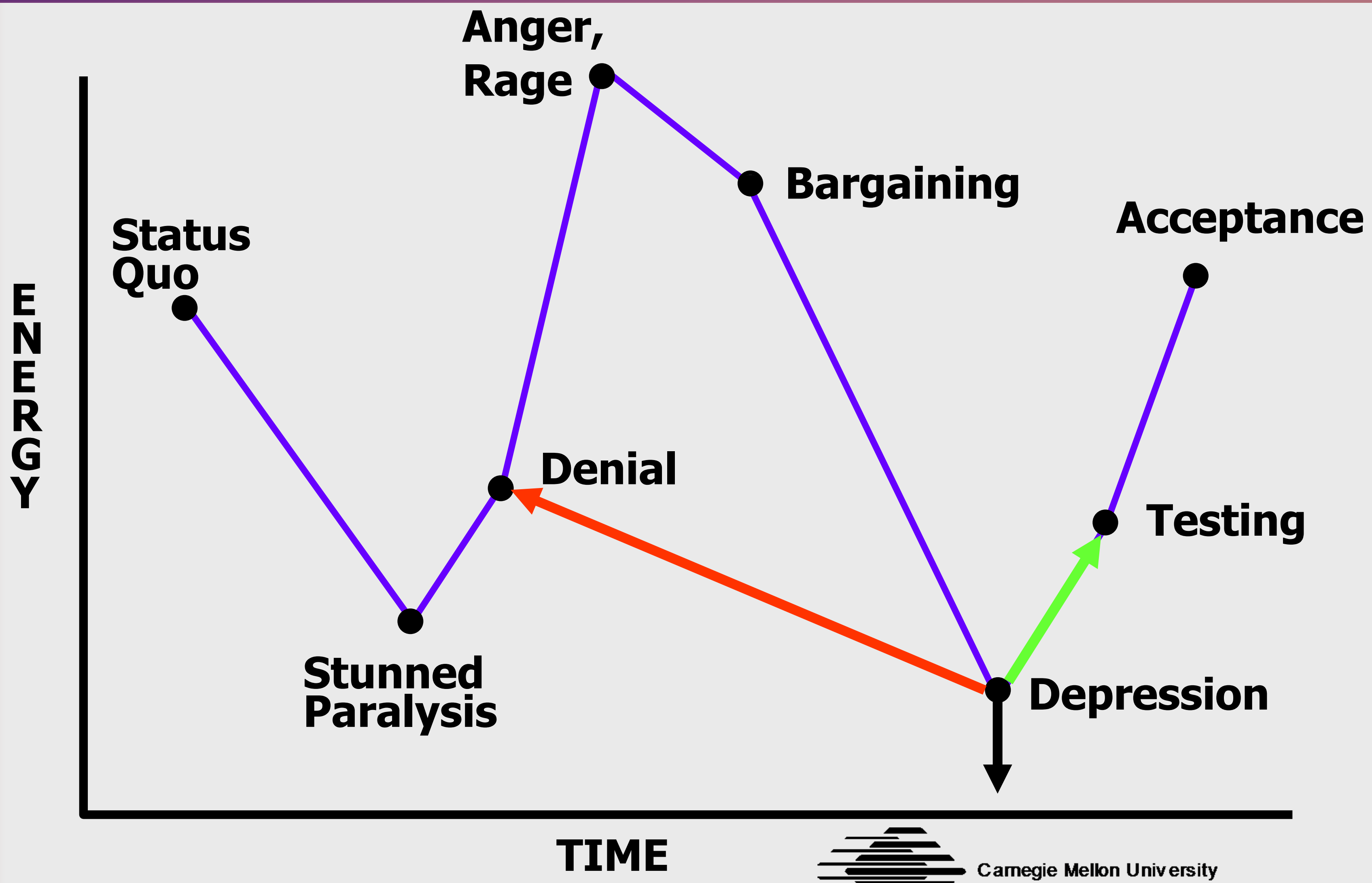
A Simple Change Model



A Sample Change Model

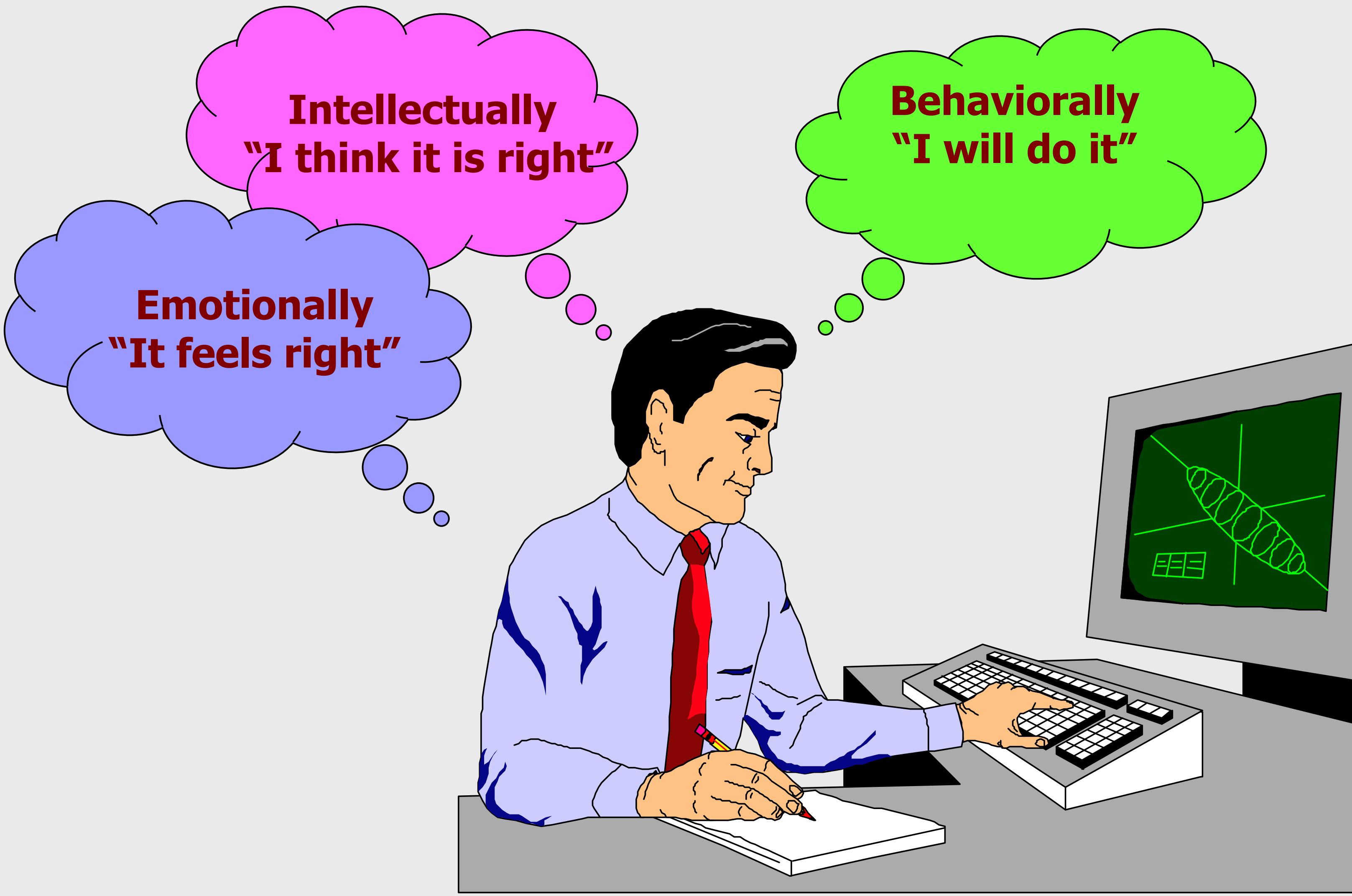


The Response to Change



Carnegie Mellon University
Software Engineering Institute

Three Ways People Respond To Change

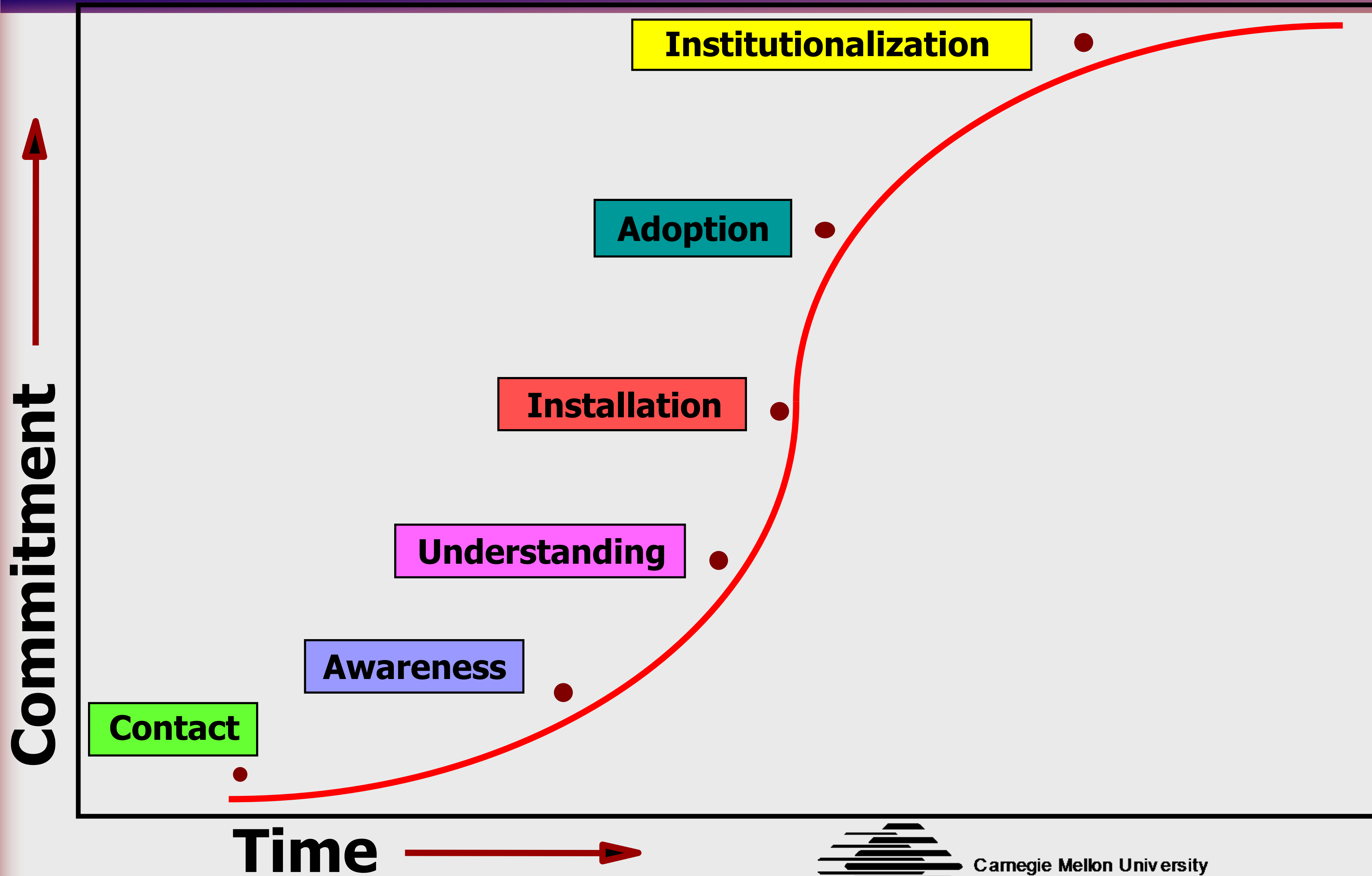


Managing Complex Change Requirements

VISION	SKILLS	INCENTIVES	RESOURCES	ACTION PLAN	CHANGE
	SKILLS	INCENTIVES	RESOURCES	ACTION PLAN	CONFUSION
VISION		INCENTIVES	RESOURCES	ACTION PLAN	ANXIETY
VISION	SKILLS		RESOURCES	ACTION PLAN	GRADUAL CHANGE
VISION	SKILLS	INCENTIVES		ACTION PLAN	FRUS-TRATION
VISION	SKILLS	INCENTIVES	RESOURCES		FALSE STARTS

"Managing Technological Change"
Carnegie Mellon University
Software Engineering Institute

Commitment is a Phased Process



Laws of Organizational Change

The "Laws" of Organizational Change

- Most teamwork involves change, and change is seldom easy
- It is unlikely that anyone will successfully change an organization without first asking its people to change as well

People Don't Resist Change

- People don't resist change – they resist being changed
- Arbitrary mandates to change normally result in people digging in their heels in resistance regardless if they recognize the change is good for them or not
- If you want cooperation ask for other's opinions:
  - What do they want to happen?
  - What do they fear?
  - What suggestions do they have to ensure the success of the effort

People Don’t Resist Change - 2

- Communicate often
- Listen more
- Seek to develop a —sharedvision” of the future state
  - Communicate clearly and regularly why things must change
  - Describe your vision for the change
  - Clearly describe the first steps to be taken
  - Link the team‘s work and the vision for change
- Seek answers to the question, —Howwill things be different?”
- How will it be determined or measured if we have changed or not?
- Link the suggested change to the business objectives

Things Are the Way They Are Simply Because They Got That Way

- Somebody wrote the policy and procedures based on their best information and understanding of the environment, competition, culture, opportunities, constraints etc.
- Somebody decided to try and follow the policies and procedures or decided not to for a personal or professional reason
- Before you attempt to change something, first take time to understand the history behind the problem

Unless Things Change, They Are Likely To Remain the Same

- If you want improvement, people will need to change the way they work
  - The change may be small and seemingly insignificant
  - The change may be large and irreversible
  - Satellite Company Example
- Avoid ―Tampering” - Overreacting to a problem or mistake without fully understanding the causes of the problem or error
  - Tampering often leads to higher costs and more errors – the opposite of what is desired for the business

Change Would Be Easy if it Weren't For All of those People

- Management would be easy if it weren‘t for the employees
- We could satisfy the requirements if the customer would just decide what it wanted and stop making changes
- Bottom Line Message – People are the organization and the organization is for the customers and end users
  - We must pay attention to the people as well as the systems or technical process we build
  - Managers play a key role in creating empowered teams or describing the key role of the project

Leading Change

- Change is a physical event so it should not be surprising that many people have strong reactions to it
- Team leaders or change agents should allow team members and others who are being asked to change to think about and come up with individual answers to the following questions:
  - What am I giving up?
  - What's in it for me?
  - How will the new process make it easier and more efficient to perform my job?

Leading Change - 2

- What information of skills do I need to be successful in the new process / environment
  - This may need to be repeated many times until people can absorb and translate the change into new tasks
- What happens if I have trouble changing?
  - Be honest!
- How do I go about making changes?
  - Developing action plans with those who must implement them goes a long ways to achieving the desired change
- How will I know how I'm doing?

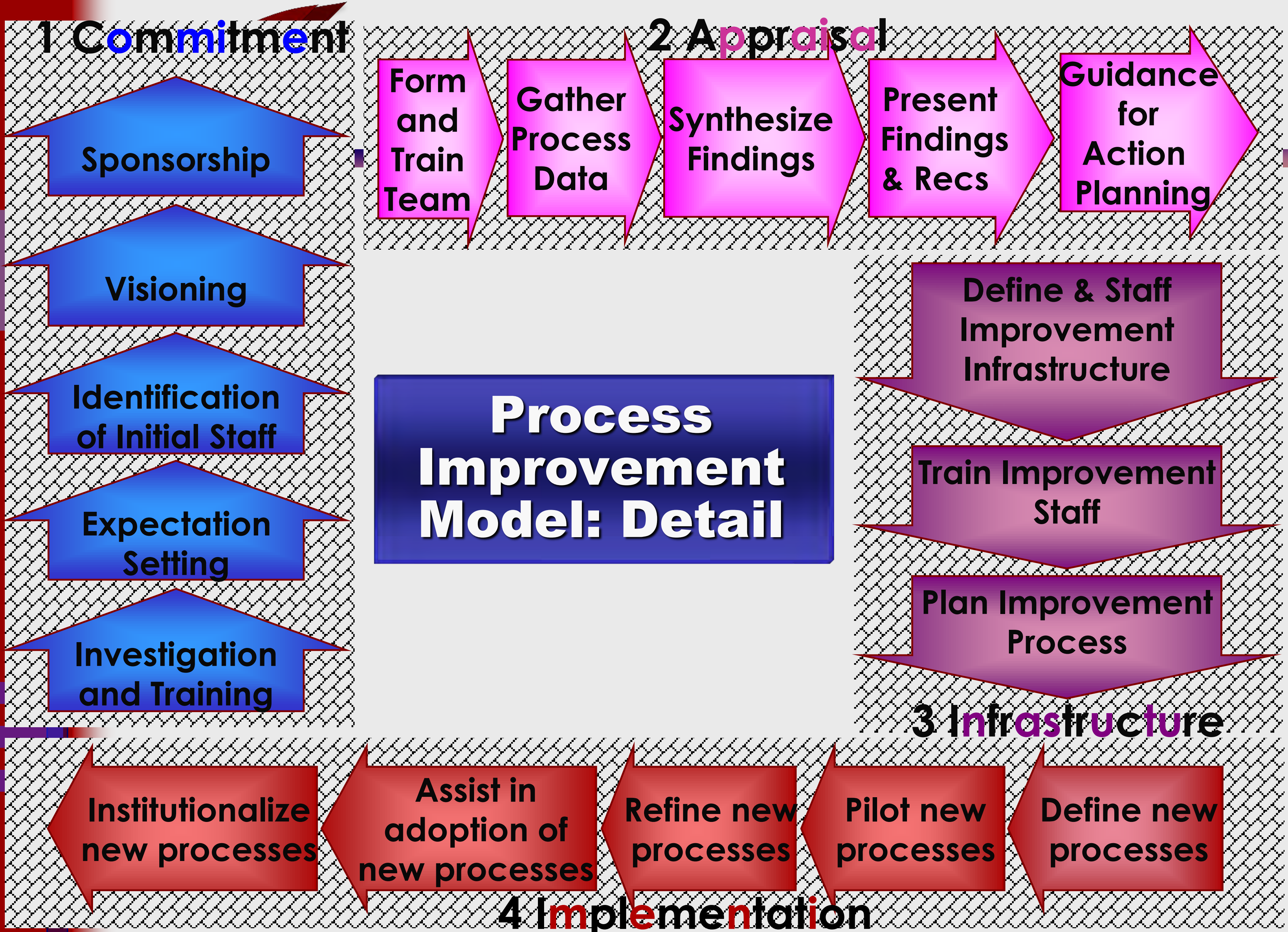
Summary
Laws of Organizational Change

- Change does not happen overnight
- People must be given sufficient time to change and supported along the way

SEPG and the Consulting Process

Module Objectives

- Discuss the SEPG as “Internal” Consultants
- Review the skill set needed by SEPG members
- Review the Six (6) Step Consulting Model proposed for internal consultants


1 Commitment
Sponsorship
Visioning
Identification of Initial Staff
Expectation Setting
Investigation and Training
2 Appraisal
Form and Train Team
Gather Process Data
Synthesize Findings
Present Findings & Recs
Guidance for Action Planning
Process Improvement Model: Detail
Define & Staff Improvement Infrastructure
Train Improvement Staff
Plan Improvement Process
3 Infrastructure
Define new processes
Pilot new processes
Refine new processes
Assist in adoption of new processes
Institutionalize new processes
4 Implementation

The SEPG as a Group of Consultants

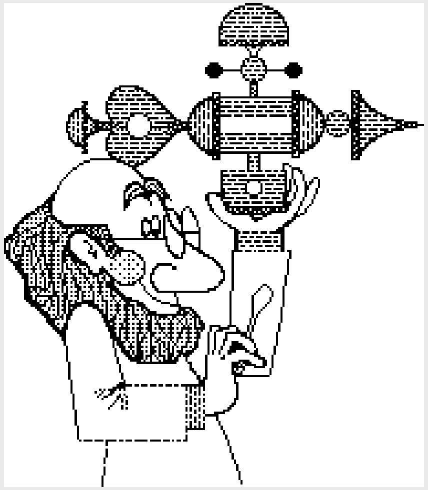
- —You are consulting anytime you are trying to change or improve a situation but have no direct control over the implementation. If you have direct control, you are managing, not consulting.”
- If you do all the work, you are under someone else‘s control
- Consulting is about having leverage and impact when we don‘t have direct control
- The SEPG consults as facilitator and collaborator

Skill Sets Needed by Consultants

- Technical Expertise
- Interpersonal Skills
- Consulting Process

Technical Expertise

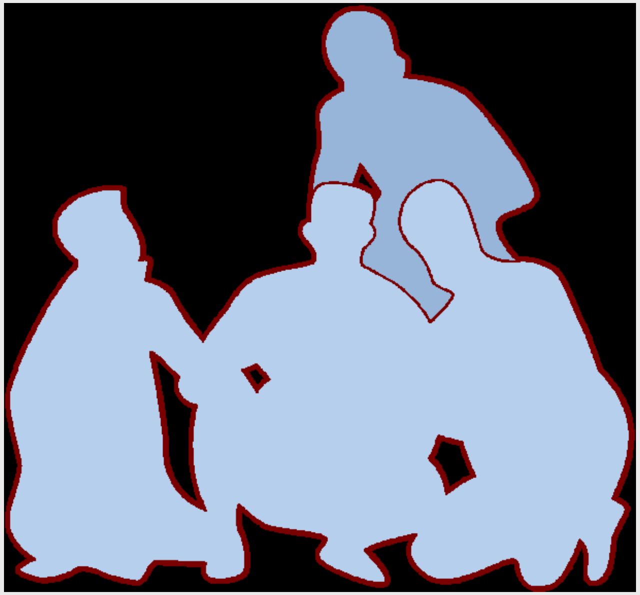
- Systems / Software management is not the same as system / software Process management.
- Useful areas of technical expertise for SEPG members:
  - Process definition and modeling
  - QA, CM, Test, Architecture, Systems Engineering, TQM, methodologies, application domains
  - Project Management including Risk Management
  - Measurement
  - Organizational behavior, systems theory



Interpersonal Skills

Effective listening

Facilitation



Team building

Meeting management

Group process

Conflict management

The 6-Step Consulting Model

Entry & Relationship Building

Contracting

Data Collection & Diagnosis

Feedback & Decision to Act

Planning & Implementation

Evaluation, Extension, Recycle, or Termination

Source: adapted from P. Block, <u>Flawless Consulting,</u> and Participant's Guide, SEI Collaborative Consulting Skills class

Entry & Sensing

- Purpose: Build the foundation for an authentic working relationship
  - Establish a trusting relationship
  - Learn what must be done to get a contract in place
- Process: Initial meetings between client and consultant
  - Understand and sense the client's expectations
- Outcome: Decision of whether you and the client are going to proceed and how you will do so

Source: adapted from Participant's Guide, SEI Collaborative Consulting Skills class

Contracting

- Purpose: Gain explicit agreement of what is expected of each other
  - Gain explicit agreement on how you and the client are going to work together
- Process: Meetings with clients, including stakeholders, and consultant
  - Make clear that you need the client's continuing support and what you can offer as the consultant
- Outcome: An explicit contract in which you agree on the project objectives/outcomes, expectations of each other, project plan or process, membership and roles, milestones, and completion dates

Source: adapted from Participant's Guide, SEI Collaborative Consulting Skills class

Data Collection

- **Purpose**: To bring together existing relevant data that will define the client's problems clearly, energize the making of appropriate decisions
- **Process**: Data gathering and analysis
- **Outcome**: Data are collected to enable the client and key stakeholders to make informed decisions about process improvement strategies.

Source: adapted from Participant's Guide, SEI Collaborative Consulting Skills class

Feedback & Decision to Act

- Purpose: To present a summary of the gathered information in a way which tells the story as you have seen and heard it
  - To create enough synergy within clients to stimulate useful problem solving and specific next steps
- Process: Presentation and decision making meeting(s) with all those who provided data
- Outcome: Decisions that shape specific intervention strategies are made by the client and consultant

Source: adapted from Participant's Guide, SEI Collaborative Consulting Skills class

Planning & Implementation

- Purpose: To gain agreement, commitment, and collaboration on the action plan
  - To build the project planning and monitoring structure to maintain constancy of purpose
- Process: Planning sessions with the client, key project members, key stakeholders, etc.
  - Education, training, and feedback sessions with all those involved
- Outcome: Resources are secured and organizational support, participation, and commitment to proceed are maintained
  - Tasks identified in the implementation plan are conducted and completed

Source: adapted from Participant's Guide, SEI Collaborative Consulting Skills class

Evaluation, Extension, Recycle, Termination

- Purpose: To gather feedback and evaluation of the consultant's behavior and the project's outcomes
  - To end or revise the client-consultant relationship
- Process: Feedback and evaluation meetings for the project and the consultant
  - Termination or contract revision meeting between client/consultant
- Outcome: For consultant, clear and concise feedback from the client's perspective on his/her effectiveness and/or contribution
  - For the organization, lessons learned for future cycles of process improvement

Source: adapted from Participant's Guide, SEI Collaborative Consulting Skills class

Conveying of Information and Experience

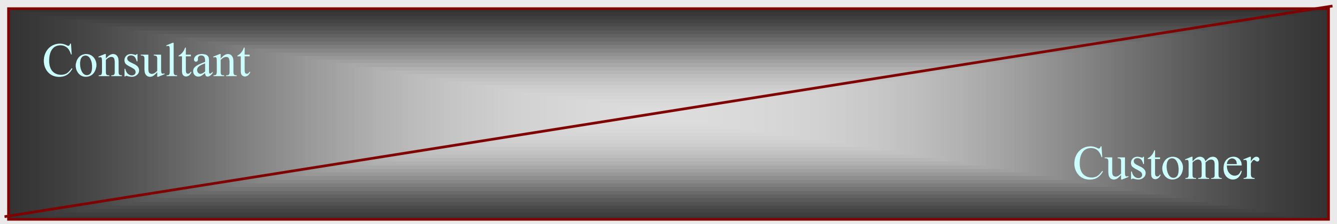
- Providing **Training** in order
  - To convey technical and organizational change concepts to individuals and groups who need to have an in-depth knowledge of the topics
  - Training is not used by itself to transfer years of experience to the participants
- Providing **Mentoring**
  - To share with a select group of individuals the psychology and philosophy behind the concepts of training or of processes, procedures, guidelines, templates, etc.
  - Mentoring sessions are set up with an Expert and up to 4 people who have been selected to be mentored
  - Experiences and war stories are shared in order to bring about a sense of reality and understanding for the Client's people that are being mentored

Conveying of Information and Experience - 2

- On the Job Experience **with Coaching**
  - For many companies, training is really reduced to On-the Job-Training.
    - This usually translates into ‗trialby fire".
  - Providing coaching of individuals and small groups while they are working on the project usually allows them to see the practicality of the ideas in their everyday life
    - If individuals and projects can see the benefits and practicality for themselves, their willingness to try out the new or revised ideas increases

Consulting Roles Are a Continuum

Responsibility



Expert

Collaborative

Observer

Consultant Role

Source: adapted from Participant's Guide, SEI Collaborative Consulting Skills class

Getting Support for Process Improvement From Above and Below

Getting Support for Process Improvement From Above and Below- Objective

- Share ideas on how one can win support for process improvement from one's employees and one's managers

Getting Support From Above and Below

- Below
  - Provide "visible" management support (not just indicate you are committed through memos - be willing to go the extra mile)
  - Be willing to provide necessary training and education and plan to attend yourself
  - Seek out your change advocates, listen to their ideas, and share your ideas with them
  - Introduce process improvement activities in bite-sized chunks. Evolutionary not Revolutionary!
  - Protect your people by making their involvement part of their job description

Getting Support From Above and Below - 2

- Below
  - Realize their productivity may decrease before it increases because they'll be trying new ideas
    - Bath tub effect
  - Encourage overt resistance. If individuals are openly protesting, encourage them to do so and really try to listen to their point of view
  - Let your people know, however, that you are personally committed to this process improvement effort and are interested in them contributing to make it successful

Getting Support From Above and Below - 3

- Below
  - Reward individuals and teams for following the processes, procedures, and standards and producing a quality product on time and within budget
  - Hesitate to reward individuals or teams for "firefighting" due to poor processes, poor planning, or poor execution
    - Story of no reward for project following process with good results
  - Hold periodic review meetings where the effectiveness of the process changes and the resulting product quality are discussed and where changes in direction may be made (not just a status reporting meeting)

Getting Support From Above and Below - 4

- Above
  - Ensure upper level mangers of your personal commitment and involvement in the process improvement effort
  - Choose a small set of metrics to collect and report that will provide real information to the upper level managers (Vic Basili - Goal, Question, Metric, paradigm)
  - Allow upper management to overtly protest
  - Try to understand what it is they need that you are not providing them
    - SEI Watts Humphrey Story – Betty Deimel

Getting Support From Above and Below - 5

- Above
  - Ask for periodic review meetings to discuss process improvement and product quality
  - Share your own project's successes/failures in implementing process improvement activities. Keep track of each participant's efforts
  - Try to understand upper management's business goals and attempt to align your project's process improvement efforts to support those goals

Hand-Holding Support

Having Multiple Personalities

- To be effective in process improvement and quality management it helps to have multiple personalities
  - Personality 1 – These are the processes and rules and YOU WILL follow them in order to achieve our process and product quality goals
  - Personality 2 – Forget about the rules, how can I help you do be successful in your current effort?
    - Evolutionary attitude

Painting A House

- First house Tim Kasse bought in Arizona - 1978
- Cowboy neighbor – hated men with long hair
- TK – no experience in painting
- Started project without significant preparation – How hard can this be?
- After 30 minutes, neighbor who was professional painter came over to explain process
- Physically took TKs hand and showed him how to properly use a paint brush – 15 minutes
- Result – House was painted, quality job that would stand up against the weather and neighbor was happy

Hand-Holding Support

- Motorola Emulator Project
  - Project behind on schedule
  - Quality Management Group provided resources to assist with Unit Test
  - Preached strict following of the software development methodology and quality activities
  - QM Engineers sat side-by-side with developers to perform Unit Testing
  - Talked to developers and developed Unit Test Plan according to organizational standard processes
  - Conducted the tests

Hand-Holding Support - 2

- Project was successful
- Vice-President was complimentary to the development team
- Development Project Manager asked Director of Quality Management if he would like to offer that support again
- NO! but we will help you understand the process we followed and support you in a collaborative way

Hand-Holding Support - 3

- Conducting Structured Walkthroughs – QM Team
  - Ensured all documents including the life-cycle work product that was to be reviewed and the associated standards were available to all reviewers
  - Did all of the training
  - Served as Moderator, Reviewer, Recorder, and Follow-up
  - Provided data analysis on major and minor defects
  - All development reviewers had to do was prepare and show up – the first time
  - Evolved from Expert to Collaborator to Observer as project members saw the results for themselves

Effective Technical Transition Strategies

Handling Non-Compliances

- Handling Non-compliances
  - Provide all non-compliances to the lowest possible level with suggestions for improvement
  - Let all levels of practitioners and managers get angry over non-compliances then tried to offer rationale and suggestions
    - Requires process and quality representatives that are highly skilled technically and in interpersonal skills
  - Escalate up to Senior Manager only if practitioner and all other levels of management rejected the non-conformance report and stated no correction would be carried out

Provide Process Improvement Advice Based on Appraisal Results Not on the Desired Level

- Naval Air Warfare Center
  - Developing software for sighting cannon on a battleship
  - 60 people
  - In the middle of a 2-year lifecycle
  - Entering Integration and Systems Test
  - Admiral in Washington DC demanding a CMMI ML2
  - Assessment results show organization is ML1 with standard weaknesses in almost every ML2 process area
  - As the External Consultant what do you advise this organization to do?

Provide Process Improvement Advice Based on Appraisal Results Not on the Desired Level- 2

- Naval Air Warfare Center - cont
  - Focus on testing techniques and offer consulting support in integration and systems testing
  - Add enough Configuration Management to control the configuration items that may change due to the testing effort
  - Add enough Requirements Management to control any late requirement change requests
  - Perform Peer Reviews on an ad hoc basis to ensure that any changes are at least reviewed before being implemented
  - Perform some Quality Assurance to ensure that these activities are being done

Testing

- Testing
  - Involve developers who are responsible for Unit Testing in reviewing the Systems Test plans and procedures
  - Invite those who conduct Unit Tests to observe the Integration and Systems Testing activities
  - Invite the Systems Testers to observe and support the developers in their Unit Testing activities

Peer Reviews

- Institutionalized use of peer reviews in Chinese corporate culture
  - Overcame cultural barrier of —losing face” when a colleague would be presented with major defects in his/her lifecycle work product.
    - It took three major attempts and 3 years of mentoring, coaching and convincing to prove “everyone” in the organization would lose face if major defects were not found and eliminated before the product was shipped
    - The CIO declared this the most significant process improvement in his Chinese culture. Hong Kong housing development board asked the Singapore IT shop to teach them Peer Reviews and provide consulting support

Peer Reviews - 2

- Institutionalized use of peer reviews in Chinese corporate culture cont.
  - Provided Peer Review Training with a Case Study
  - Provided extra training for Moderators
  - Served as —coach” of a Peer Review and intervened throughout the face-to-face part of the Peer Review
  - Videotaped Peer Review sessions with coaching
  - Provided two additional Peer Review trainings with coaching over the 3 years
  - Finally got people to admit their unwillingness to submit major defects and cause their colleague to lose face
  - Convinced developers and managers that —everyone” in the organization would lose face if major defects were not found and eliminated before the product was shipped

Configuration Management

- Configuration Management
  - Support project or developmental configuration control from the organizational control group if projects are too small to have their own Configuration Management Representative
  - Help the transition from project control to organizational control at the designated points in the lifecycle
  - Help the Project Manager to keep control on the evolving configuration items
    - Keep excellent change history records from which to issue periodic and on-demand Configuration Status Accounting Reports

Configuration Management - 2

- Show PM how understanding of the frequency of work product changes can lead to the decision to use formal reviews such as Inspections or Structured Walkthroughs versus Informal Walkthroughs or Buddy Checks
- Provide baseline or milestone configuration audits to show Project Managers their project is meeting all requirements and approved requirements change requests and that all necessary hardware and software components plus corresponding documentation are reviewed and available or are in the process of being developed
  - Functional Configuration Audits
  - Physical Configuration Audits

Measurement

Establish Measurement Objectives

- While establishing measurement objectives, a project/organization should:
  - Document the purposes for which measurement and analysis is done
    - What is the information needed?
    - What questions are you answering with the data?
    - How will the measurements affect project behavior?
  - Specify the kinds of actions that may be taken based on the results of the data analyses
  - Continually ask the question – what value will this measurement be to those people who will be asked to supply the raw measurement data and who will receive the analyzed results – —Why are we measuring this?”
  - Maintain traceability of the proposed measurement objectives to the information needs and business objectives
- Ensure business objectives and measurement objectives are developed with clear —Whys” this measure will support the business and quality goals of the project and organization

Information Needs

- Information needs typically reflect:
  - Management needs
    - Established management objectives (Reduce errors found by customer)
  - Technical needs
    - Recurring technical problems
  - Project needs
    - Increase accuracy of estimation (Planning)
    - Increase performance (Project performance constraints)
  - Process improvement needs
    - Increase effectiveness of requirements elicitation process
  - Product needs
    - Reduce defect density of delivered software
  - Customer requirements information needs
    - Increase ability to meet customer requirements

Measurement Objectives

- Based on the —informatiomeeds” derived Measurement Objectives for either the organization and/or the project may include:
  - Reduce time to delivery based on historical data indicating late delivery
  - Deliver specified functionality completely
  - Improve prior levels of quality
  - Improve levels of profit (keep project within or below budget)
  - Improve prior customer satisfaction ratings

Measures

- Measures in line with these measurement objectives may include:
  - Normalized time in hours and tenths of an hour (actual time, size, and complexity)
  - Delivered functionality as a percentage of the functional requirements
  - Normalized defect density as the number of defects per 1000 lines of code
  - Normalized costs within stated limits
  - Customer satisfaction ratings based on averaged and normalized surveys

Quantitative Measures

- Example Measurement Objectives for either the organization and/or the project with more emphasis on quantitative measures include:
  - Reduce time to delivery to a specified percentage
  - Reduce total lifecycle costs of new products by a percentage
  - Deliver specified functionality by a specified increased percentage
  - Improve prior customer satisfaction ratings by a specified percentage compared to past ratings
  - Improve prior levels of quality by reducing the number of defects of type A that get shipped with the product OR
  - Improve prior levels of quality by reducing the number of defects of type A that get shipped with the product without exceeding the delivery date by more than 10% and the budget by more than 8%

The ability to reach and then predict reaching these quantitatively specified goals will increase as the organization increases in its process capabilities

Good to Best Practices

- Best Practices
  - Seek good processes on existing projects and making them best practices for all projects throughout the organization
  - Motorola Microsystems Story of Adapting Assembly Language Coding Standards from a successful Project Manager

Criticality

- Criticality
  - Provide the strongest hand-holding support for critical projects to the organization and to those who want that help
  - Ensure the success of each project that you work with and —circle the wagons” on the other projects that do not want to cooperate

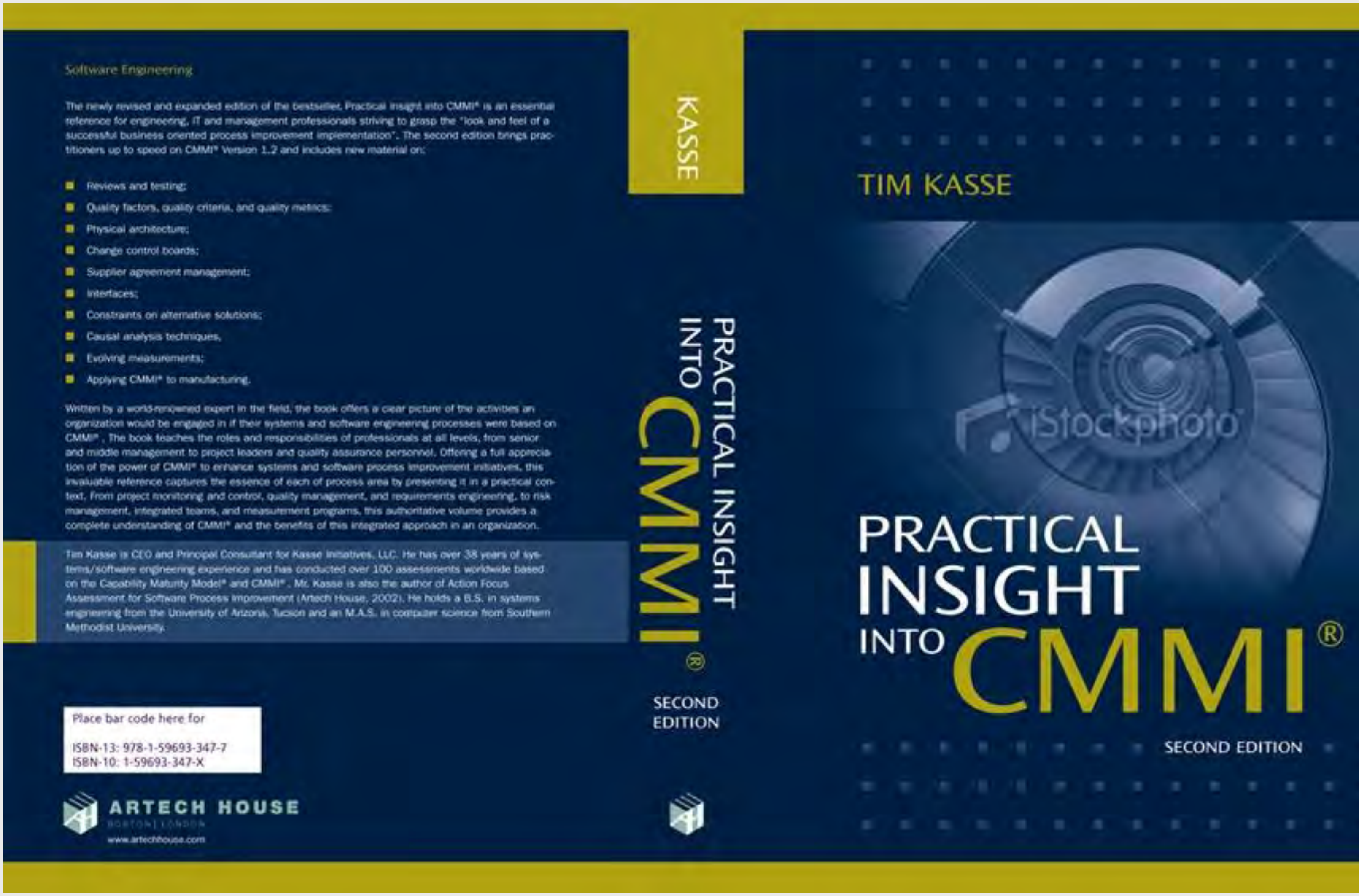
Summary

- Process improvement and quality management is not something that can be dictated in a memo or a "all hands" speech and then expected to happen
- Good processes become best practices when the projects see that they can be used and achieve required process and product quality results
- People, projects, and organizations will change and continue to change if they see the results and see the benefit for themselves!
- The only high-probability way to get processes to be followed and people to change is to provide "hand-holding" support until those that are being supported see that benefit for themselves

Tim Kasse

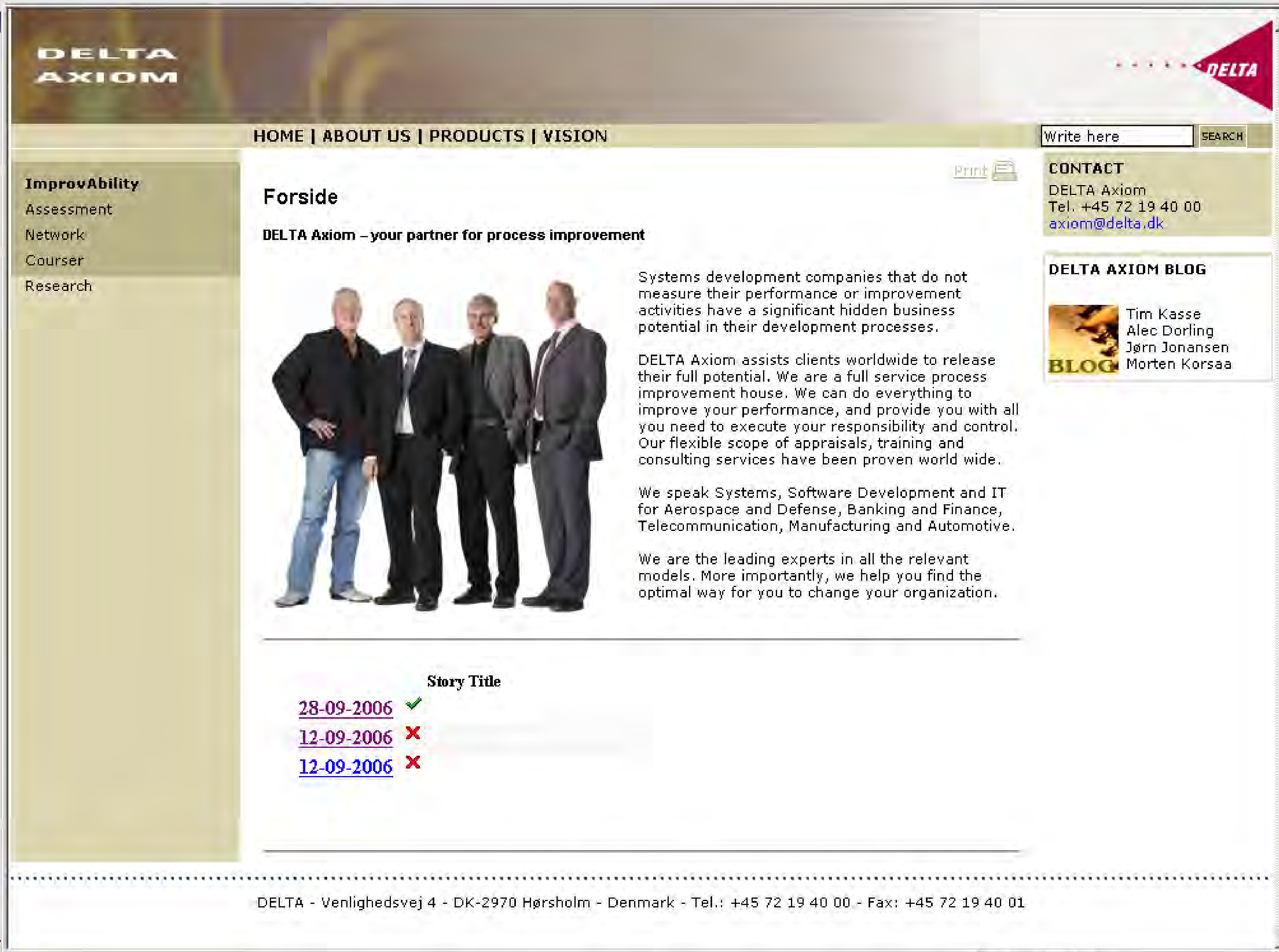
- CEO and Principal Consultant of Kasse Initiatives
- Visiting Scientist - Software Engineering Institute
- Visiting Fellow - Institute for Systems Science / National University of Singapore
- Author of Action Focused Assessment for Software Process Improvement
- Author of Practical Insight Into CMMI

Books
From Kasse Initiatives



Books
From Kasse Initiatives - 2

Delta Axiom


DELTA AXIOM
DELTA
HOME | ABOUT US | PRODUCTS | VISION
Write here
SEARCH
ImprovAbility
Assessment
Network
Courser
Research
Print
Forside
DELTA Axiom – your partner for process improvement
Systems development companies that do not measure their performance or improvement activities have a significant hidden business potential in their development processes.
DELTA Axiom assists clients worldwide to release their full potential. We are a full service process improvement house. We can do everything to improve your performance, and provide you with all you need to execute your responsibility and control. Our flexible scope of appraisals, training and consulting services have been proven world wide.
We speak Systems, Software Development and IT for Aerospace and Defense, Banking and Finance, Telecommunication, Manufacturing and Automotive.
We are the leading experts in all the relevant models. More importantly, we help you find the optimal way for you to change your organization.
Story Title
28-09-2006
12-09-2006
12-09-2006
CONTACT
DELTA Axiom
Tel. +45 72 19 40 00
axiom@delta.dk
DELTA AXIOM BLOG
BLOG
Tim Kasse
Alec Dorling
Jørn Jonansen
Morten Korsaa
DELTA - Venlighedsvej 4 - DK-2970 Hørsholm - Denmark - Tel.: +45 72 19 40 00 - Fax: +45 72 19 40 01

Method Park

Bernd Hindel
CEO & Founder of Method Park

Tim Kasse – CEO & Principal Consultant
Kasse Initiatives LLC

STORM (Strategic Technology and Operational Risk Management)

Innovative Approach for Organizational Integrated Risk Management Approach

Kobi Vider – Picker

K.V.P Consulting

Kobi.Vider@hotmail.com

+972522946676

Intellectual Property Disclaimer

- The content of this presentation, STORM / IRMA / HERMES models and all related materials are the intellectual property of Kobi Vider and K.V.P Consulting
- THE CONTENT ON THIS WEBSITE IS PROVIDED "AS IS" WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND EITHER EXPRESSED OR IMPLIED. TO THE FULLEST EXTENT POSSIBLE PURSUANT TO APPLICABLE LAW, Kobi Vider and K.V.P Consulting DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, IMPLIED WARRANTIES OR CONDITIONS OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE,
- All materials contained in this presentation and all models related documentation are protected by copyright laws, and may not be reproduced, republished, distributed, transmitted, displayed, broadcast or otherwise exploited in any manner without the express prior written permission of Kobi Vider and K.V.P Consulting
- Without a written approval from Kobi Vider or K.V.P Consulting; you are not permitted to modify the Content. You are not permitted to make use of the Content for commercial gain. You must keep intact all copyright notices for the Content and provide, reasonable to the medium or means you are utilizing, the name of Access Copyright and the title of the Content if supplied. Such credit may be provided in any reasonable manner

Agenda

Background to the Need

- Critical facility emergency events and incidents are managerial, not technical
- Mission and objective statement as much as other, must include quantitative objectives that are stated in a clear way
- Basic building block is the capability to accurately evaluate the unit's effectiveness along with the efficiency of its resource usage
- The main challenge is to integrated the overall **risks** in the '**spider net**' and to **understand** their **true impact**

K.V.P Consulting

Multi Disciplines Surrounding (MDS)

Known

Unknown

Known
Unknown

Unknown
Unknown

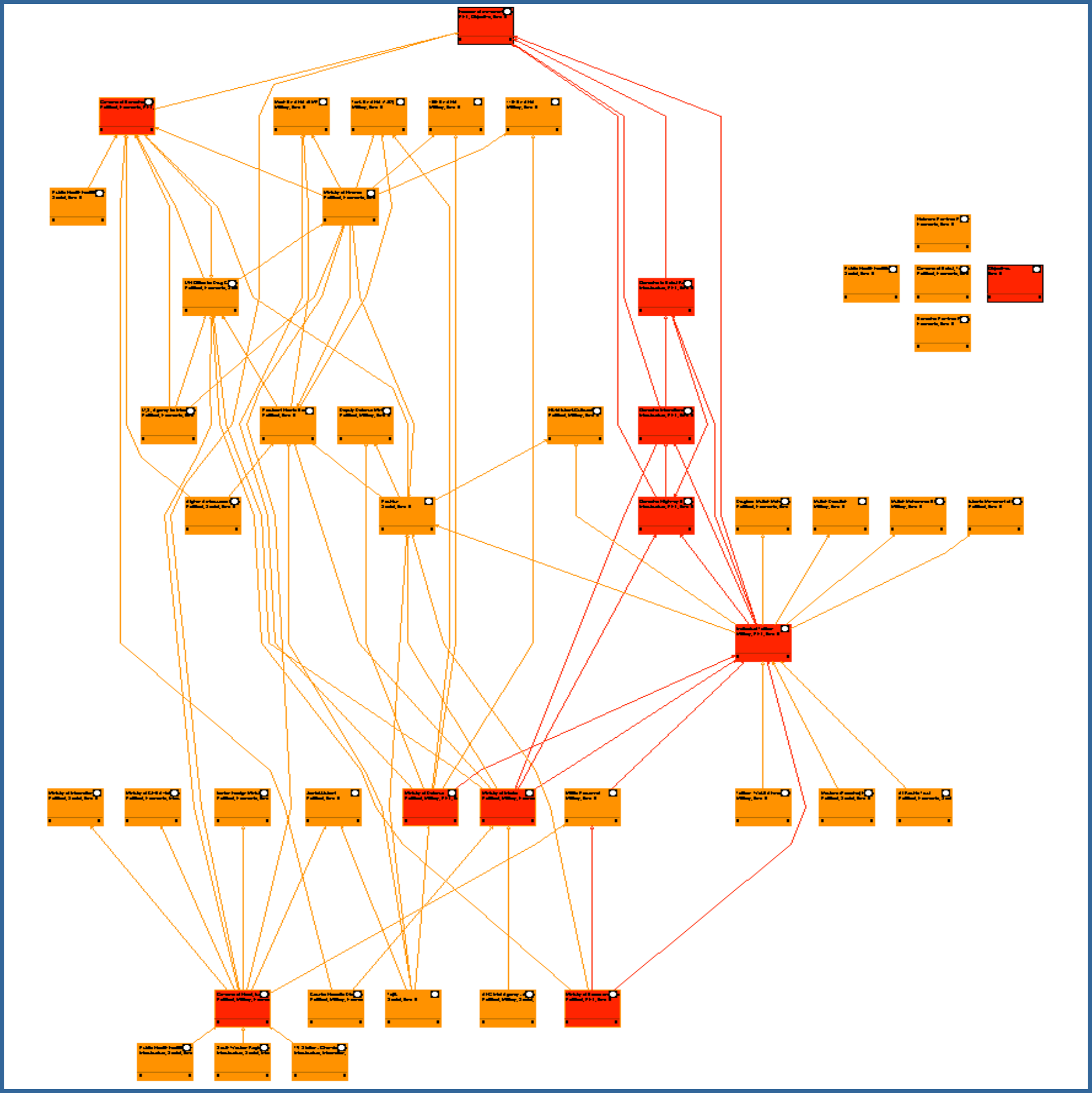
Known
Known

Unknown
Known

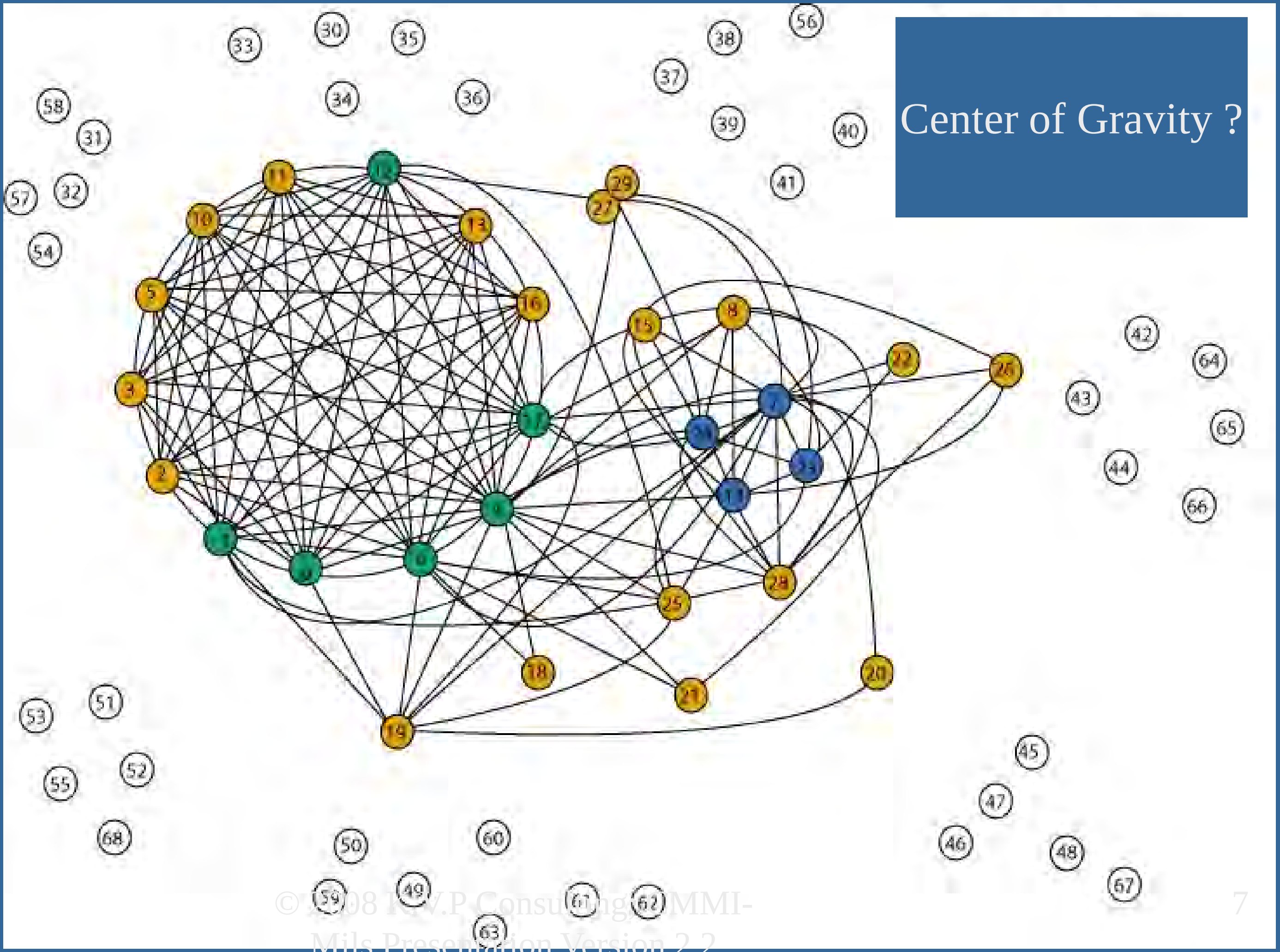
Known

Unknown

A Complex Effects-based Environment



Military Combat Services Support Challenges in the Battlefield



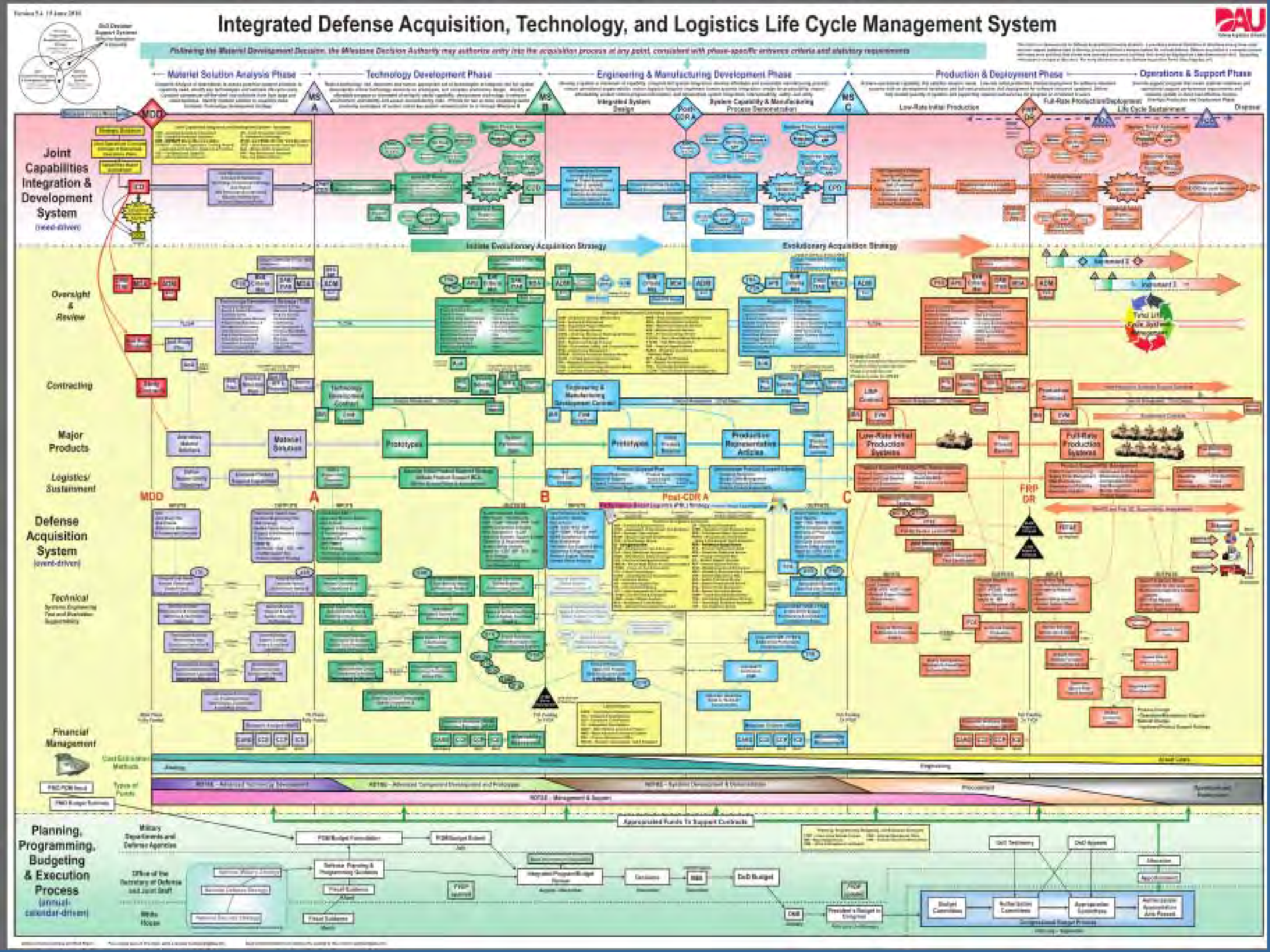
Work Assumptions

- Decisions are managerial, not technical
- Objective statement as much as other, must include quantitative objectives that are stated in a clear way
- Basic building block is the capability to accurately evaluate the unit's effectiveness along with the efficiency of its resource usage

K.V.P Consulting

Multi Disciplines Surrounding (MDS)

Typical Lifecycle Description



Conceptual Case Study


LOG
FIN
Admin
PMO
Eng
Production
IT SEC
R&D
Storage
SVC
© 2010 Sanborn
© 2010 Europa Technologies
© 2010 Tele Atlas
© 2010 Google
©2007 Google™

K.V.P Consulting

Background

- A key to organizational wisdom is
  - judgement and decision making,
- Which requires an understanding of the complexity of a situation, but also requires the ability to make sense and simplify a situation or event so that appropriate and effective action can be taken.
- Three important drivers for the development of organizational wisdom are
  - Experience
  - Passion to learn, and
  - Culture.
  - Processes for acquiring organizational wisdom such as transformational leadership, organizational culture and knowledge transfer are also part of our focus and will be discussed.

The Challenge Statement

Organizations that need to establish business relationships with other businesses face major challenges including:

- The need for creating a win-win-situation
- The effort to align business processes and link up information systems across company borders

Organizations do not know how to efficiently use interoperability from the business perspective to identify the fundamental artifacts that are related to business interoperability

Common Failures - 1

Organizational Crisis are predominantly managerial, not technical.

- Lack of defining business objectives in quantitative terms and structure
- Inadequate definition of 'Good Enough' level
- Inability to differentiate different business objectives and success factors for the different domains and lifecycle phases
- Inadequate resource usage and adjustment to Plan and Objectives
- Failure to identify and manage risks
- Poor or mismanaged service / operational requirements
- Uncontrolled baselines, no configuration management
- Misunderstood business / operational needs and objectives

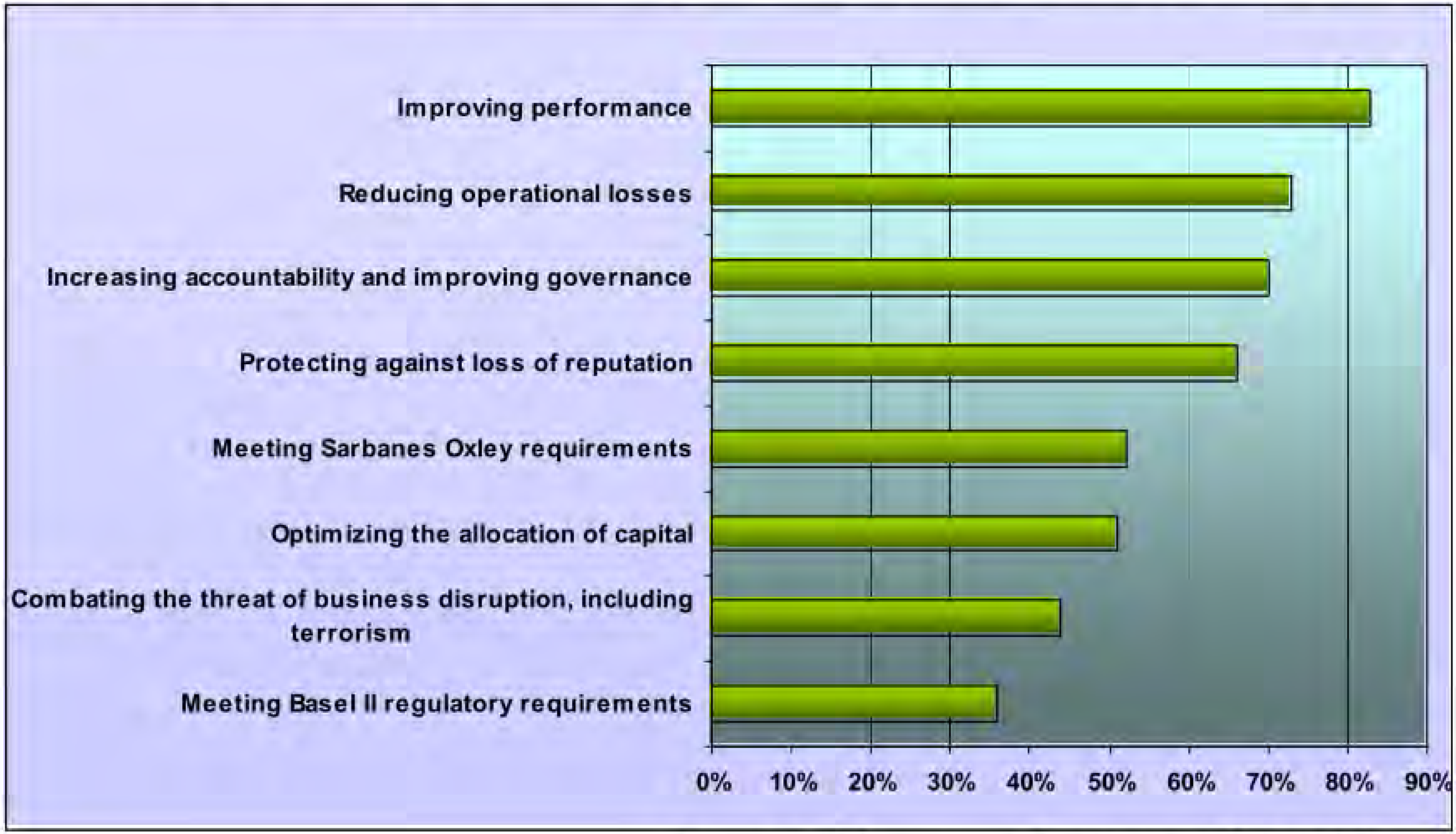
Common Failures - 2

- Poor contractor acquisition or management
- Lack of skills, capability and training
- Poor planning and tracking
  - Value Stream
  - Equipment
  - Resources
  - Finance
- Poor / misuse of data and measurements
- Inability to estimate accurately
- No quality assurance / control
- Poor communications

Cost Demonstration
Power
OPP
FIN
OPP & Mang Expose

Main Areas and Response for Risk Management Improvements

Figure 1. Main Reasons to Invest in Operational Risk Measurement and Management



Source: Risk Management Association (RMA). 2003.

The Operational Need

- Management capability level from both professional and knowledge level
- Performance and reporting norms
- Self management and self discipline maintaining personal professional and knowledge capabilities
- Individual and team discipline
- Cooperation and knowledge and resource sharing
- Appropriate visibility of information, data and capabilities
- Quality of readiness and preparedness for performing mission

The Operational Need

- Centralized resource management and appropriate utilization and usage of it
- Multidimensional management (future planning, unit strategy, short term objectives, the immediate objectives)
- Initiating, developing and implementation management of new tactics and technologies
- Balanced planning and deploying new tactics improvements and new technologies in a measured way that will quantify the improvement vs. expectations
- Information, data and communication security

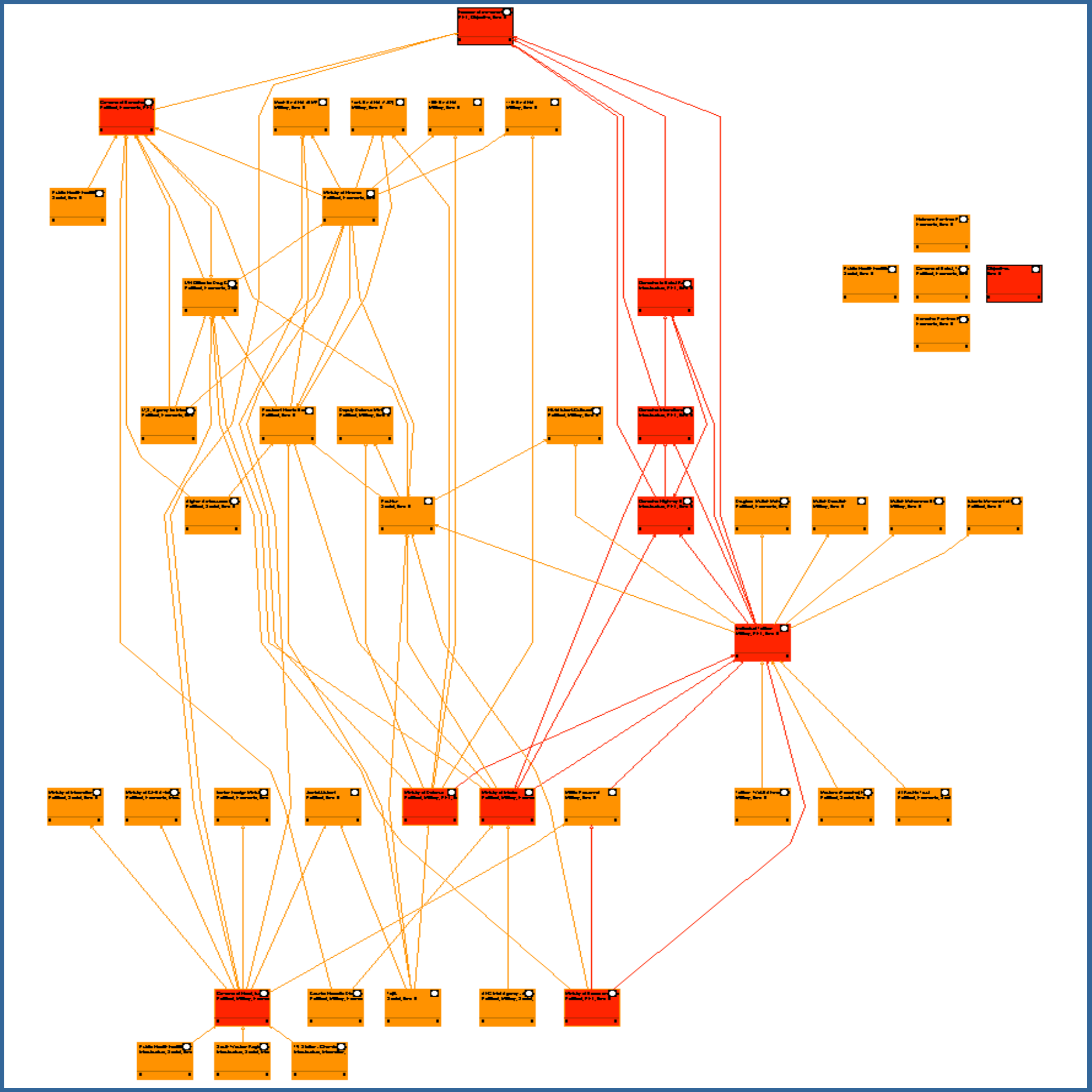
The Operational Need

- Each person working in the implementation organization will need to do the following:
  - Access the response doctrine descriptions
  - Understand all the response doctrines at a top level
  - Understand in detail the response doctrines that he or she performs
- **In addition, managers must do** the following:
  - Understand all the response doctrines at a top level
  - Understand the leadership response doctrines change management in detail
  - Understand how to lead the unit using the new response doctrines
  - Access historical measurement data for all response doctrines versions performance
  - Support implementation of new response doctrines in their own surroundings
  - Remove roadblocks to implementation

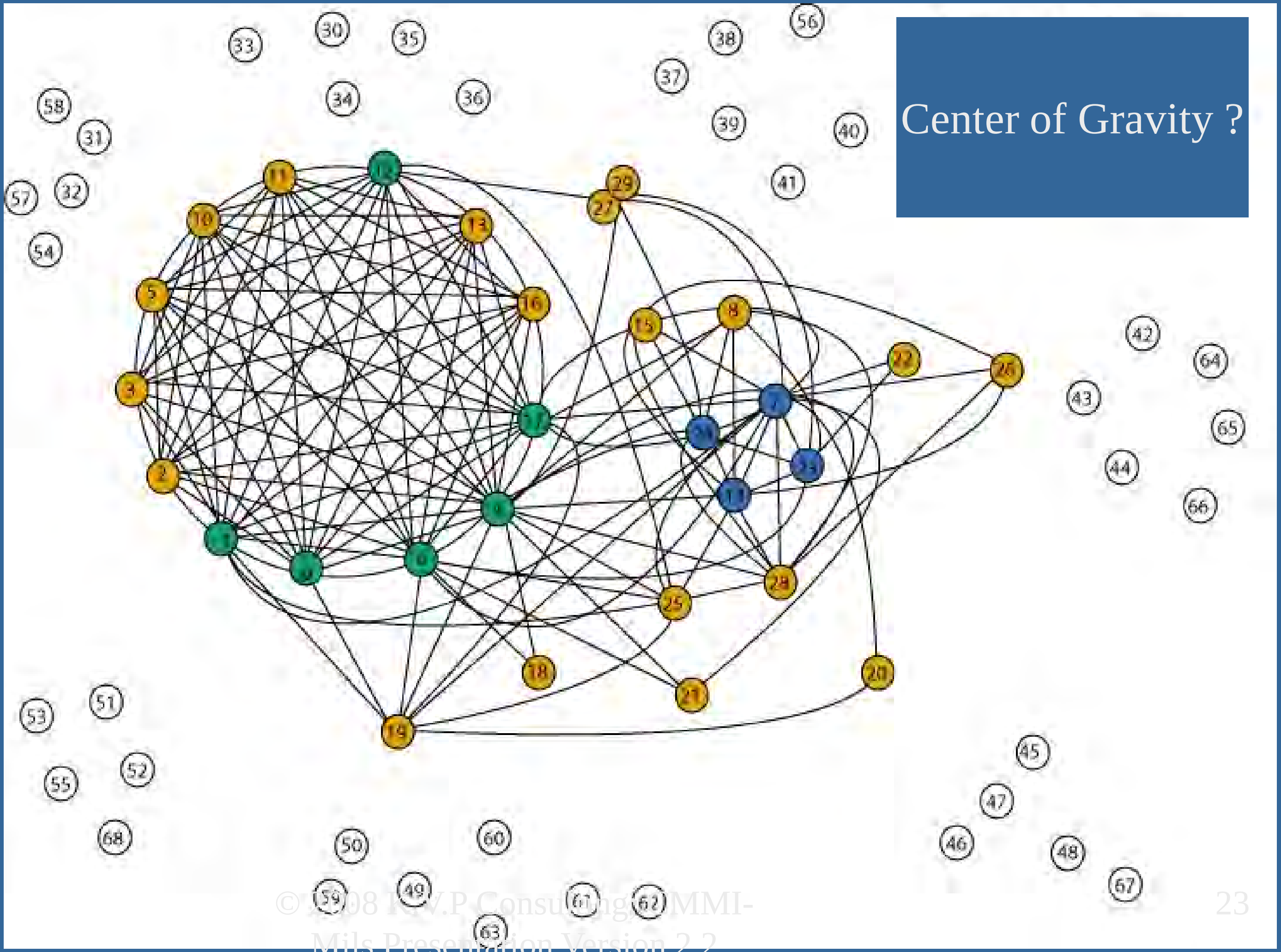
The Operational Need

- Many of these challenges were an is addressed on and **ad-hoc basis**, usually with specialized solutions or technologies that were limited to functional areas of the operational scenario or a unit that is currently in the frontline at a given time

A Complex Effects-based Environment



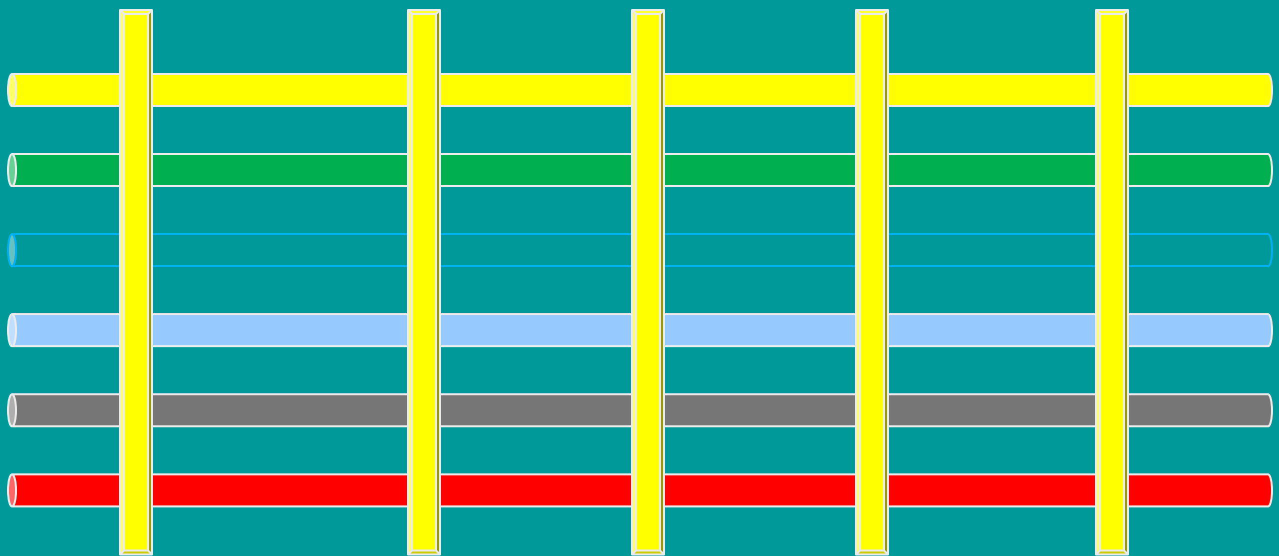
Military Combat Services Support Challenges in the Battlefield

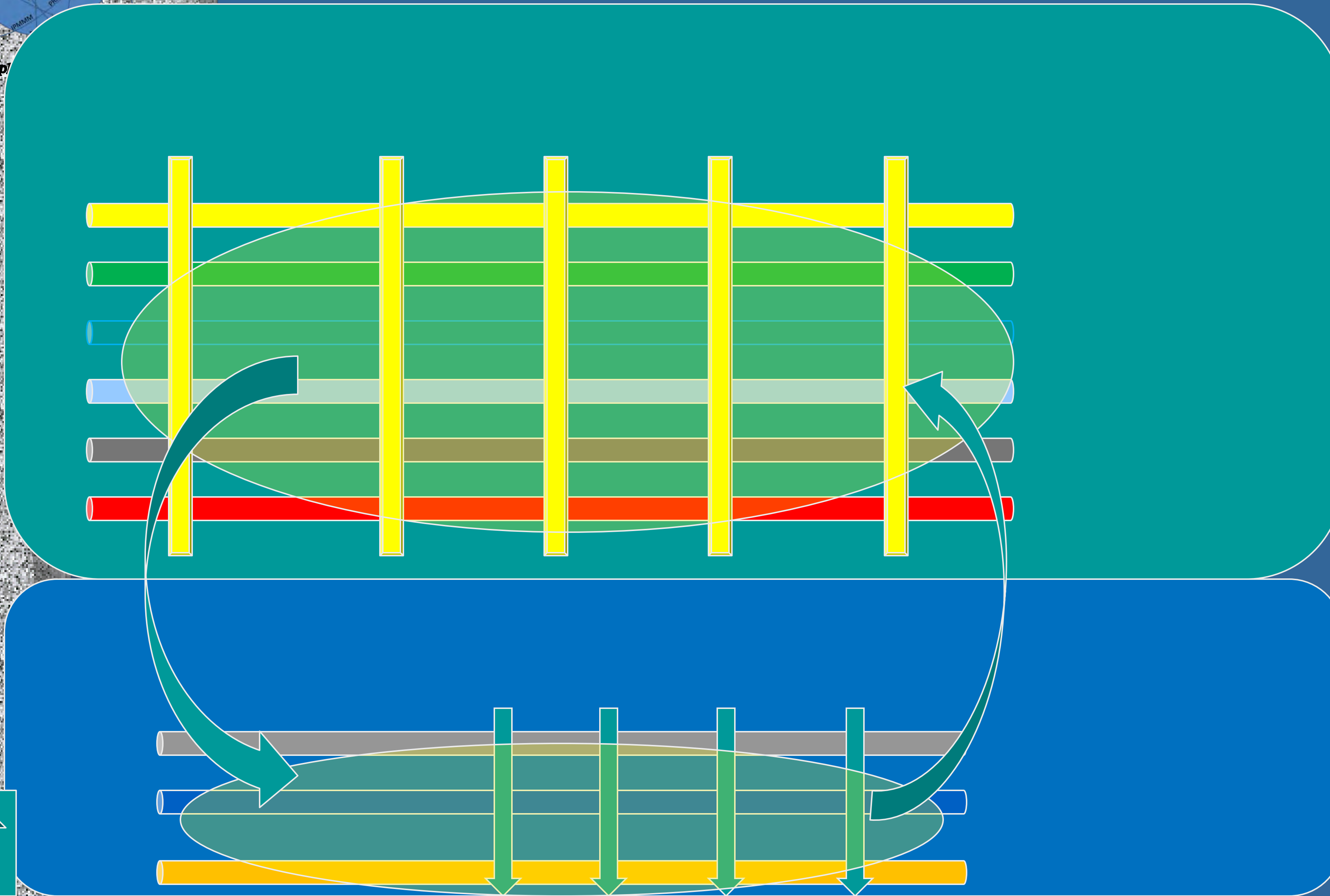


Common

The Entity

The Unit





Main Failures and its Related Cost

Date	Type of Firm	Loss (in USD)	Brief Description of Allegation
Nov -85	Bank	4 million	Computer problems with Fed payment connection
Feb-93	Corporate	1.04 billion	Unauthorized futures trading
Apr-94	Brokerage Firm	350 million	False profits reported for two years
Sept-95	Bank	1.1 billion	30,000 unauthorized trades over 11 years
Feb-96	Bank	1.3 billion	Losses from NIKKEI futures hidden in 88888 account
Jun-96	Bank	1.8 billion	Unauthorized copper trading – futures, etc.
Aug-96	Fund	19.3 million	Deal allocations delayed for personal profit
Sep96	Bank	750 million	Dummy companies used to avoid compliance
Mar-97a	Bank	130 million	Option volatilities used to inflate prices
Mar-97b	Bank	100 million	Funds transfer to personal account

Table 1: Example financial losses attributed to operational risk

Main Failures and its Related Cost

Example of Multiple Linear Regression

Month	Number of Operational Losses	Amount of Losses	Overtime in Hours	Number of Transactions	Number of System Failures
January	84	1,600,000	80	1230	41
February	93	1,893,452	110	1280	43
March	68	1,356,318	50	812	35
April	110	2,321,725	160	1523	62
May	49	1,000,987	14	710	18
June	151	2,300,012	218	1510	83

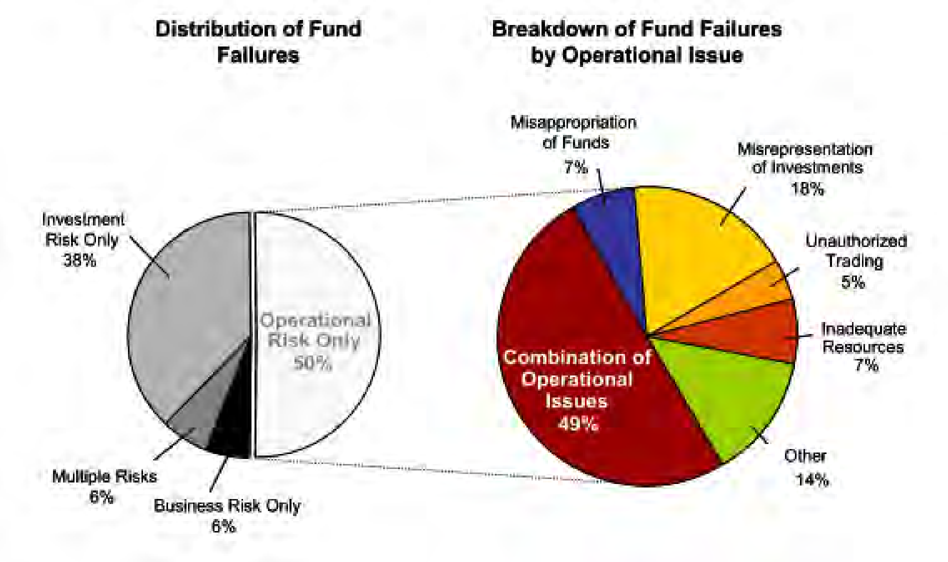
Main Failures and its Related Cost

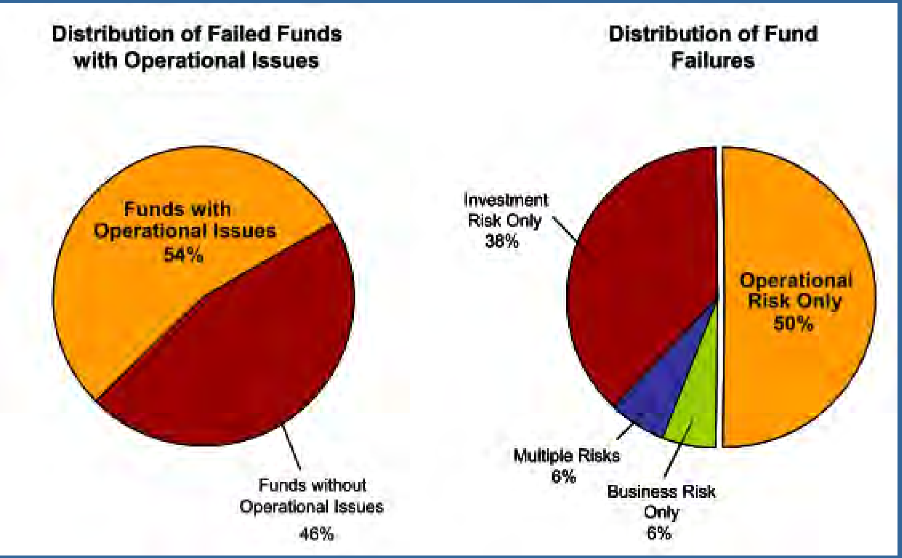
Table 1. Major North American Power Outages 1965 -2003

Event	Date	MW loss	People Affected
Northeast Blackout	Nov. 9, 1965	20,000 MW	30 million
New York City Blackout	July 13, 1977	6,000 MW	9 Million
West Coast Blackout	Dec. 22, 1982	12,350 MW	5 million
West Coast Blackout	July 2-3, 1996	11,850MW	2 million
West Coast Blackout	Aug. 10, 1996	28,000MW	7.5 million
Upper Midwest Blackout	June 25, 1998:	950MW	152,000
NE and Canada Blackout	Aug. 14, 2003	61,800MW	50 million

Source: US-Canada Taskforce report (2004)

Main Failures and its Related Cost





Main Risks Areas and Impact (Example Only)

Risk Class	Risk Type	Activity or Event	Examples	Mitigation	Frequency & Severity
People	Internal	Unauthorized Activity Lack of skilled personnel	Rogue Trading High employee turnover	Partially insured	
People	External	Fraud	Theft	Partially insured	
Systems	Internal	Model Risk	Model/Methodology error Mark-to-model error	Technical risk audit Improve quality of models/people	
Systems	External	Technology Risk	Telecommunication failure Blackouts	Contingency planning Insurance	
Processes	Internal	Transaction Risk	Execution error Settlement error Documentation/contract risk	Improve processes	
Asset damage	Internal	Physical asset risk	Pipeline Rupture Production loss Unexpected plant outage	Partially insured Contingency planning	
Asset damage	External	Physical asset risk	Uninsured or irrecoverable loss or damage to assets	Insurance	

The Challenge

- This situation where the organization is running
  - separate process improvements on different parts of the system / product lifecycle
  - With partial overall view in interactions and handshakes between these groups is introducing inefficient usage of
    - resources,
    - expensive maintenance of duplicate infrastructures
    - and Organizational Sets of Standards Processes as well as assets,
  - May result in less quality and impacting the competitive edge with their global counterparts.

The Approach to the Solution Concept

- Best practices in the model focus on activities for providing quality services to the customer and end users
- To identify improvement targets in main lifecycle areas such as operations, information, governance, people and organizational structure, portfolios, project execution, and finance
- Select processes that are critical to the system success such as stakeholder management, technical interfaces and integration

The Approach to the Solution Concept

- Build an action plan composed from the following main steps
  - Organizational map
  - Functional team and groups size and role in the lifecycle
  - Full lifecycle map
  - Setting improvement targets
  - Gap analysis
- Suggesting to the senior management to address the lifecycle and process (as a whole) as a complex of crossing services and to add additional content to the lifecycle map (as a layer) and content in the guideline that will define the different interactions as services

The Conceptual Solution

- Building on contingency theory, it outlines a comprehensive framework suggesting a fit between the level of Mission interoperability and environmental as well as internal contingencies.
- Moving from the current environment of basic process and way of thinking toward a more controlled and measured process to reduce the overwhelming amount of information that build decisions

The Conceptual Solution

- We have found that Maturity Models and practices combined with some other industry standards and methods as a new integrated approach can be used as tools to leverage procedures to support the Critical Facility and the Critical Facility al Mission objectives and capability, readiness and preparedness to achieve Mission improvement and excellence.
- It is the premise of this presentation to give you brief idea on the model concept and context. It will provide you the basic information regarding the value added by using it and how to appropriate to do it while implementing and defining it to your own Mission context

The Conceptual Solution - 1

- Building on contingency theory, it outlines a comprehensive framework suggesting a fit between the level of business interoperability and environmental as well as internal contingencies.
- Moving from the current environment of basic processes and way of thinking toward a more controlled and measured set of processes to reduce the overwhelming amount of information that is now required to build decisions

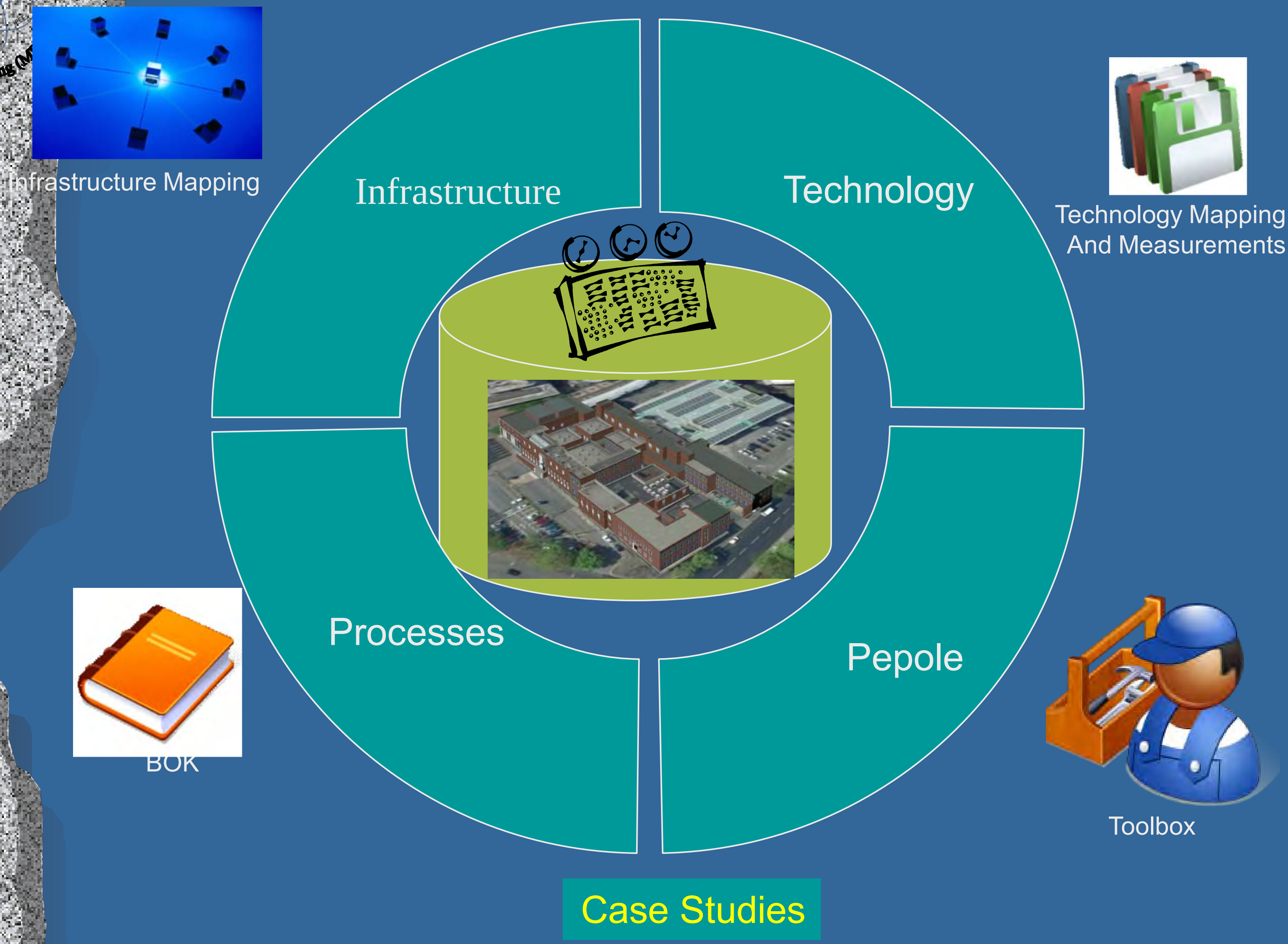
The Conceptual Solution - 2

- We have found that Maturity Models and practices combined with some other industry standards and methods as a new integrated approach can be used as tools to leverage procedures to support the organization and the organizational business objectives and capability, readiness and preparedness to achieve business improvement and excellence.
- It is the premise of this presentation to provide a brief idea on the model concept and context.
  - This presentation will provide you the basic information regarding the value added by using the model and how to appropriately interpret the model while implementing and defining it to your own business context

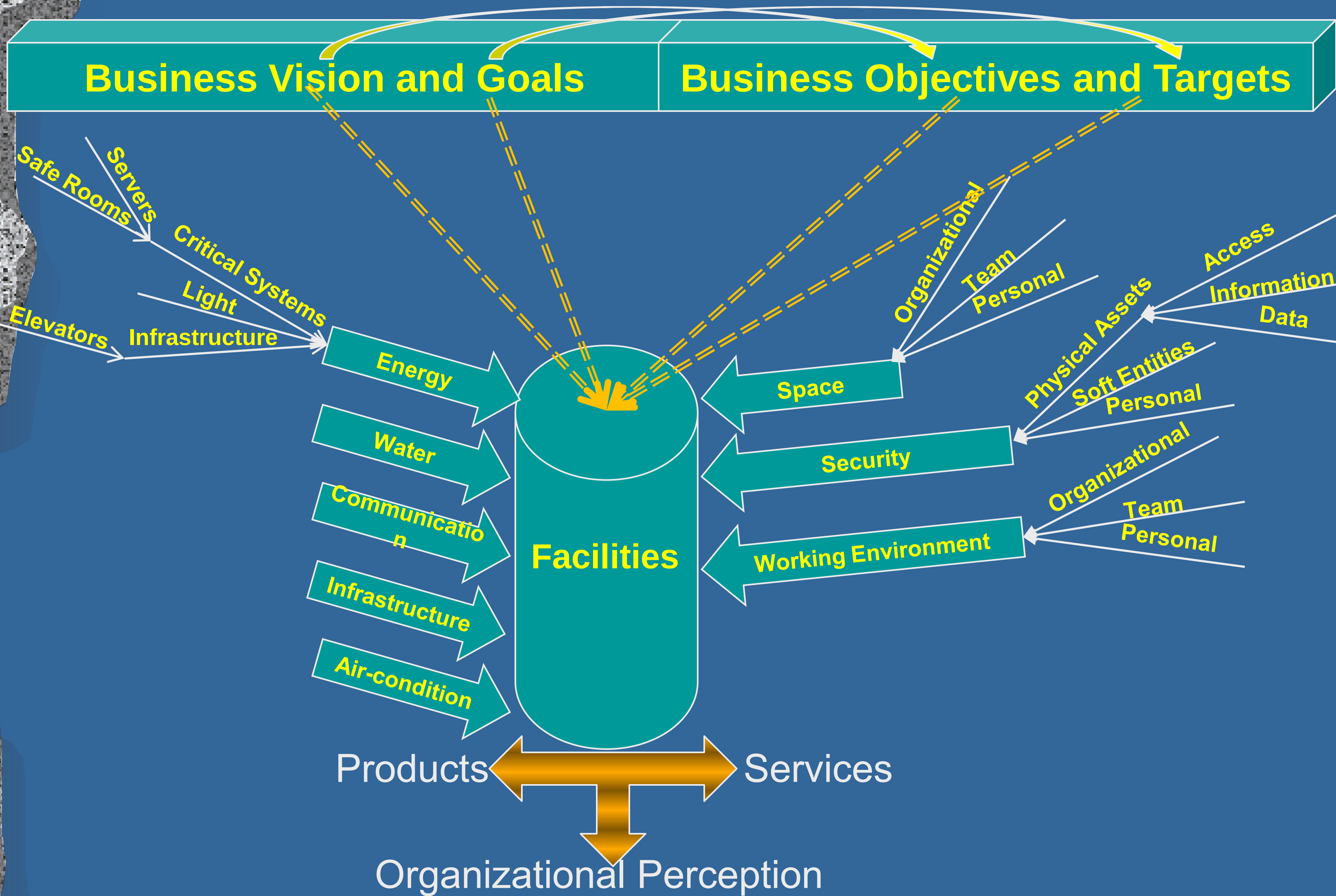
The Four Main Entities and Their Role

- Facility
  - Provide the ‘hard and physical’ working environments and infrastructure
- Technology
  - Provide the ‘soft and intangible’ working environments and infrastructure and tools
- Process
  - Provide the working procedures and instructions, which assume to guide in the most effective way how to use the facilities and technology to achieve the business objectives by the people
- People
  - Provide the individuals that build the teams within the organizational units and groups, that perform the tasks and activities described in the process

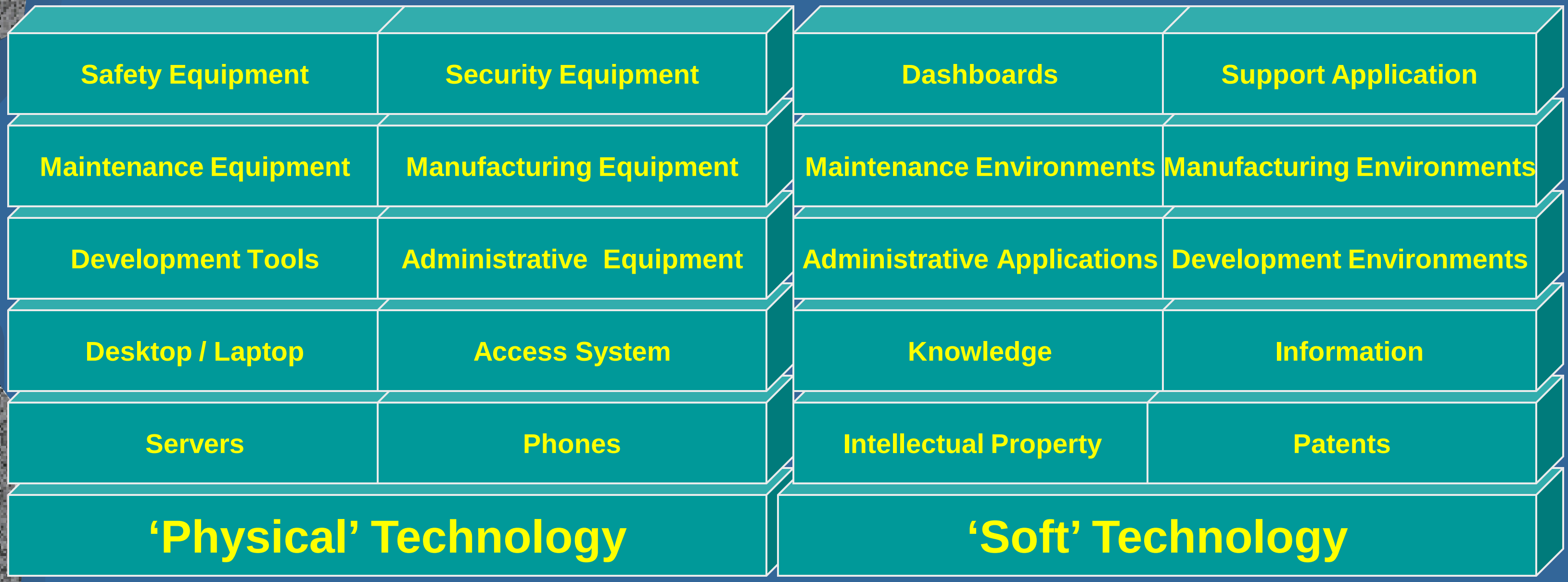
Layers Conceptual Structure



The Organization Managed Layers – Facility (as illustration only)



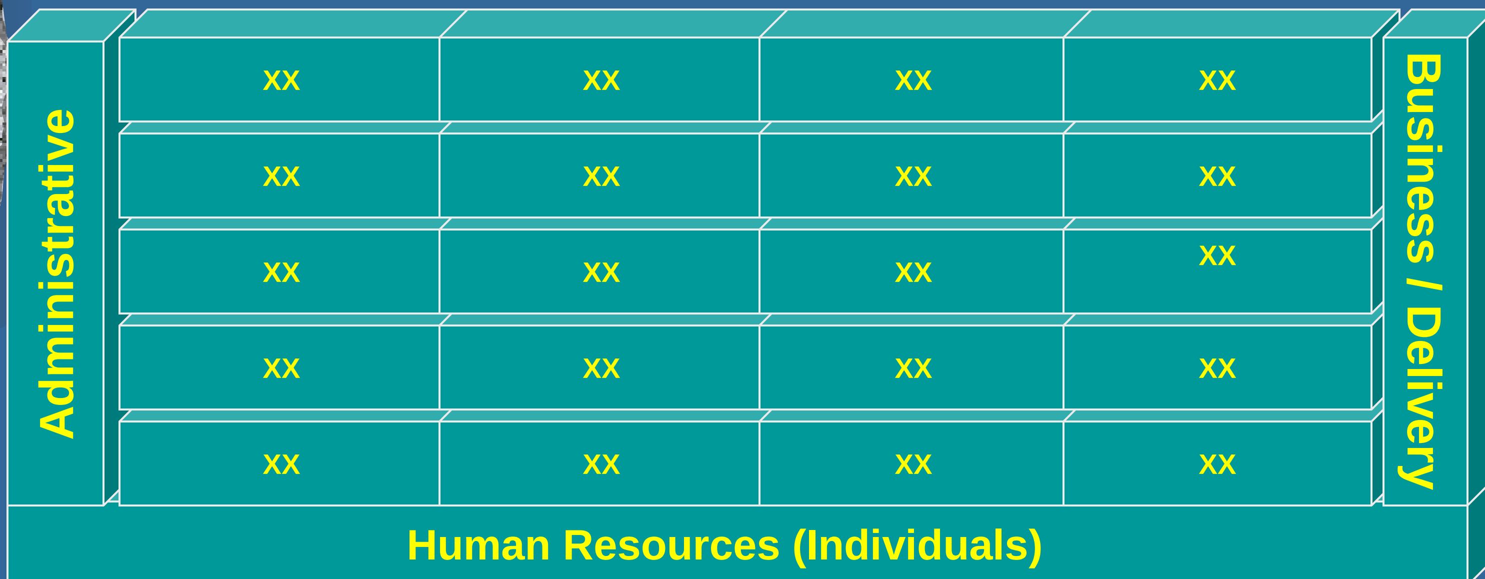
The Organization Managed Layers – Technology (as illustration only)

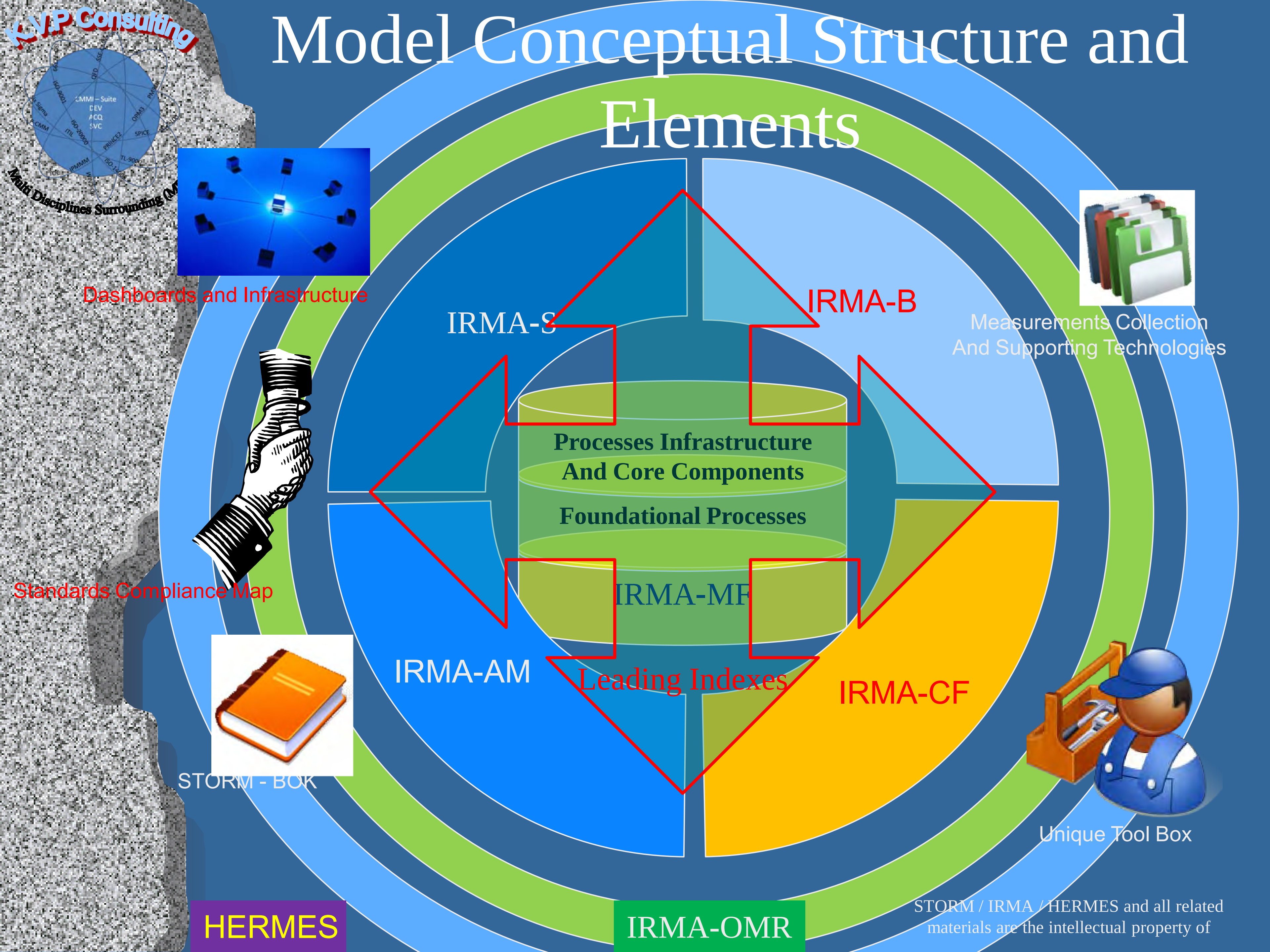


The Organization Managed Layers – Processes **(as illustration only)**

Administrative (Corporate 'wise')		Business / Delivery (Product 'wise')	
XX	XX	Acquisition / Procurement	Support
Work Environments	Safety	Maintenance	Manufacturing
Ethics	Environmental	Development	Managerial (Portfolio)
Human Resources	Security	Knowledge	Managerial (Program)
Legal	Finances	Intellectual Property	Managerial (Project)

The Organization Managed Layers – People **(as illustration only)**





Model Conceptual Structure and Elements
K.V.P Consulting
Multi Disciplines Surrounding (M
Dashboards and Infrastructure
IRMA-S
IRMA-B
Measurements Collection
And Supporting Technologies
Processes Infrastructure
And Core Components
Foundational Processes
IRMA-MF
Standards Compliance Map
IRMA-AM
Leading Indexes
IRMA-CF
STORM - BOK
Unique Tool Box
HERMES
IRMA-OMR
STORM / IRMA / HERMES and all related materials are the intellectual property of

Method Content (Model Wise)

LSPI - Light Security Performance Index – this approach is a light version of the full model that allow a unit / organization to evaluate its security procedures against known and unknown threats by using a numerical scale to compare variables (the unit performed practices) with reference constants (the LSP Index items), the objectives of LSP is to give the unit general idea on gaps in its USP (Unit Standard Procedures)

- MDSPI/MSPI - Managing Defined Security Performance Index/ Managing Security Performance Index - this approach is a higher level and more advanced method to the LSP version. This index is built on the LSP gap mapping and adding additional layer. This additional layer allow the unit / organization to evaluate its security procedures not just against known and unknown threats like LSP, but also adding the organizational view that all units using the same procedures by using a numerical scale to compare variables (mapping all units performed practices) with reference constants (the MDSPI/ MSPI Index items), the objectives of MDSPI/ MSPI is to give the unit general idea on gaps in its USPI (Unit Standard Procedures Implementation)
- SSPI - Statistical Security Performance Index - this approach is a higher level and more advanced method to the MDSPI/ MSPI version. This index is setting the foundation to understand the unit / organization practice performance by understanding the statistical behavior of it. The objectives of SSPI is to give the unit general idea on gaps in its UOPPB (Unit and Organizational Practice Performance Behavior)

Method Content (Model Wise)

- IRMA-CF - Integrated Risk Management Approach Core Foundation, is the basic model that is the mandatory Body Of Knowledge (BOK) to all other models
- IRMA-B - Integrated Risk Management Approach for Business, this is a preset and preconfigured model that address the needs the common industry companies
- IRMA-S - Integrated Risk Management Approach Security, this is a preset and preconfigured model that address the needs the security industry and agencies (e.g. secured facilities, police, fire fighters)

Method Content (Model Wise)

- IRMA-CF - Integrated Risk Management Approach Critical Facility, this is a preset and preconfigured model that address the needs the critical facilities (e.g. power plants, ports, air ports)
- IRMA-AM - Integrated Risk Management Approach Area Management, this is a preset and preconfigured model that address the needs for managing an area (geographic or defined as critical area (e.g. disaster zoon, government offices)
- IRMA-OMR - Integrated Risk Management Approach Operational Mission Readiness, this is a preset and preconfigured model that address the needs for a mission performance readiness and capability alignment

Method Content (Model Wise)

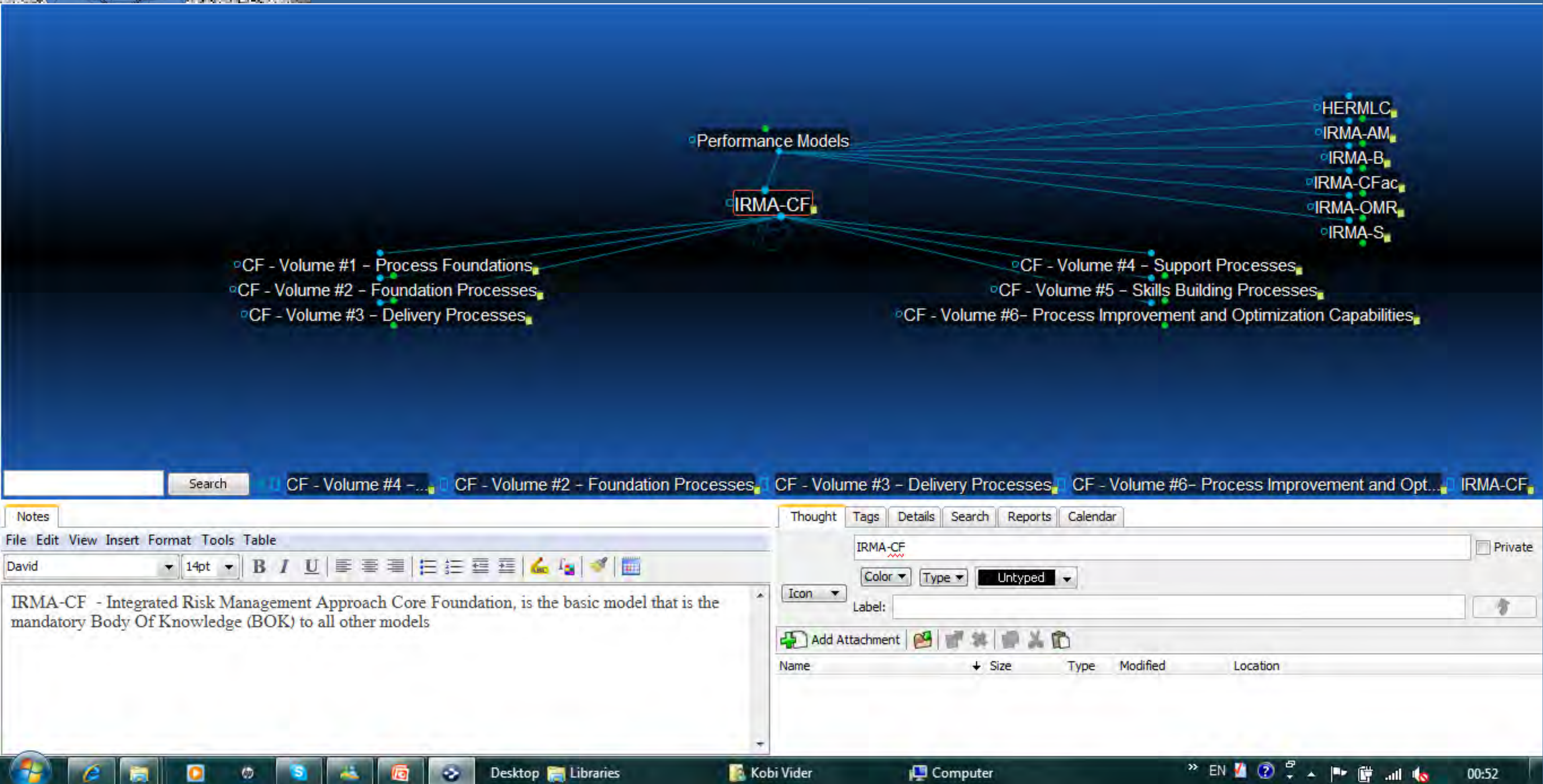
- HERMES (Harmonized Enterprise Risk Management Evaluation Standard – this standard is built from:
  - Standard Description Document (SDD)
  - Mandatory Evaluation Plan (MEP) with tailoring guidelines and preconfigured sets to address the five models
  - Interpretation Guidelines Sets (IGS) addressing the five models
  - Detailed scoping and rating scheme
- ERPI – Environmental Risk Performance Index - this approach is a light version of the full model that allow a unit / organization to evaluate its Environmental Risk analysis and management life cycle procedures against known and unknown threats by using a numerical scale to compare variables (the unit performed practices) with reference constants (the ERPI Index items), the objectives of ERPI is to give the unit general idea on gap in its USP (Unit Standard Procedures)
- HERMLC – Harmonized Environmental Risk Life Cycle - the model objectives is to address the system / product lifecycle and process as a whole with complexity of crossing services. And to enable effective and efficient analysis from the first phases the level of Environmental Risk.

Solution Structure

- Model Architecture
- Model Publication Volumes
- Model Processes

Model Architecture - 1

- STORM is a comprehensive model that covers all business and operational aspects of the organization
- It is true that the model view serves as the start point for the single individual; however the best benefit from the implementation is gained at the:
  - Corporate and division level for the business and overall operations efficiency
  - Department and Group level in their own operations (it also depends on the task and objectives statements)
  - Projects and product lines level
  - Functional groups level (e.g. security)


Performance Models
IRMA-CF
HERMLC
IRMA-AM
IRMA-B
IRMA-CFac
IRMA-OMR
IRMA-S
CF - Volume #1 – Process Foundations
CF - Volume #2 – Foundation Processes
CF - Volume #3 – Delivery Processes
CF - Volume #4 – Support Processes
CF - Volume #5 – Skills Building Processes
CF - Volume #6– Process Improvement and Optimization Capabilities
Search
CF - Volume #4 –...
CF - Volume #2 – Foundation Processes
CF - Volume #3 – Delivery Processes
CF - Volume #6– Process Improvement and Opt...
IRMA-CF
Notes
File Edit View Insert Format Tools Table
David
14pt
IRMA-CF - Integrated Risk Management Approach Core Foundation, is the basic model that is the mandatory Body Of Knowledge (BOK) to all other models
Thought Tags Details Search Reports Calendar
IRMA-CF
Private
Color Type Untyped
Icon
Label:
Add Attachment
Name Size Type Modified Location
Desktop Libraries
Kobi Vider
Computer
EN
00:52

Model Architecture - 2

Preface
Part One – About the Model
1. Introduction
2. Model Components
3. Working with the Model
4. Relationships Among Areas
5. Implementation Guidelines
6. Interpretation Guidelines

Part Two – Model Body

1. **Volume 1 – Process Foundations**
2. **Volume 2 – Foundation Processes**
3. **Volume #3 – Delivery Processes**
4. **Volume #4 – Support Processes**
5. **Volume #5 – Skills Building Processes**
6. **Volume #6– Process Improvement and Optimization Capabilities**

Part Three – The Appendices and Glossary
References
Acronyms
Glossary

Model Volumes

- Volume 1 – Process Foundations
- Volume 2 – Foundation Processes
- Volume #3 – Delivery Processes
- Volume #4 – Support Processes
- Volume #5 – Skills Building Processes
- Volume #6– Process Improvement and Optimization Capabilities

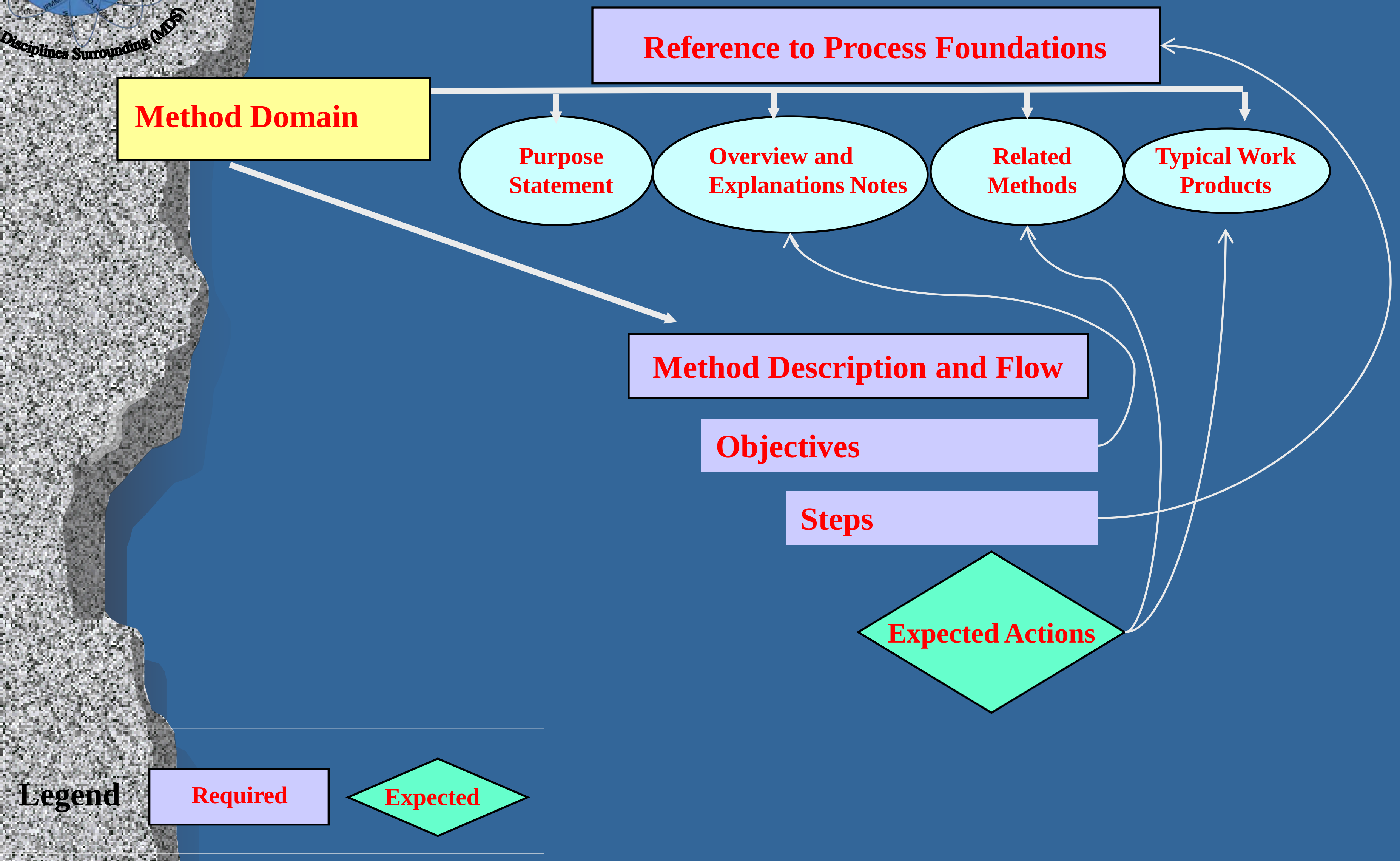
Model Architecture - 3

- The OBO-PI addresses the organization as a separated whole. For this reason we have divided it into different volumes:
  - Volume 1 – Process Foundations - this collection of practices identify the quality ingredients and requirements that are needed to establish and maintain strong and solid process
  - Volume 2 – Foundation Processes - this collection of process and practices address the requirements to develop and maintain (cradle to grave) work planning and control skills and capabilities
  - Volume #3 – Delivery Processes - this collection of processes and practices address the requirements to develop and maintain (cradle to grave) appropriate working and development skills and capabilities including work environment (tools)

Model Architecture - 4

- Volume #4 – Support Processes - this collection of processes addresses the requirements to develop and maintain appropriate support capabilities (cradle to grave) with full alignment with the organizational objectives and goals
- Volume #5 – Skills Building Processes - this collection of processes addresses the requirements to develop and maintain appropriate and efficient procedures to enable effective skills building that will answer the organizational need
- Volume #6– Process Improvement and Optimization Capabilities - this collection of processes and practices addresses the requirements to develop and maintain appropriate process understanding to enable focused optimization capabilities with full alignment to the mission objectives and goals

Volume Chapter Structure



Additional Supporting Informative Components

- There is further information that is provided in the form of the following components:
  - Examples
  - Amplifications
  - References
  - Notes

K.V.P Consulting

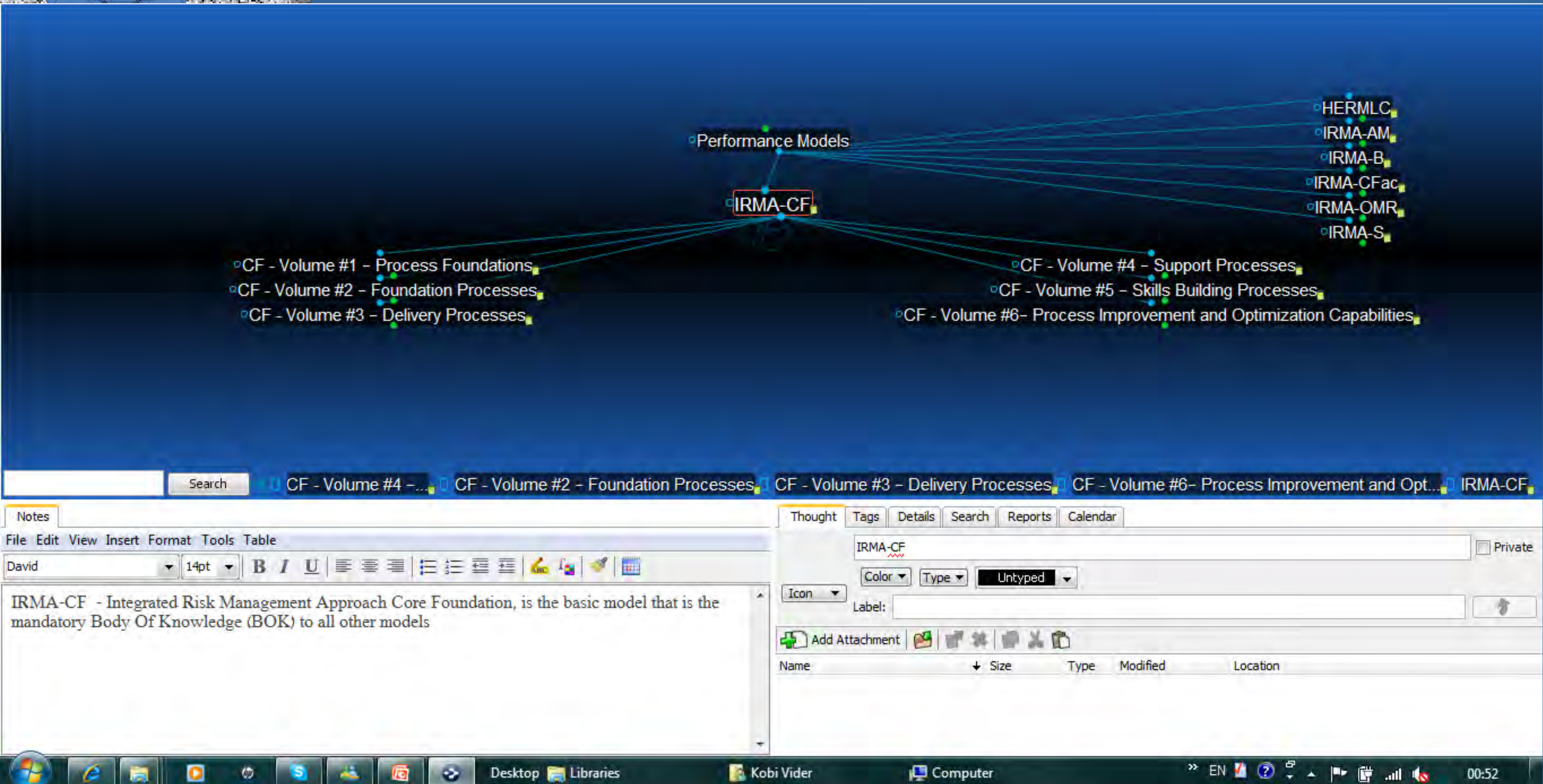
Multi Disciplines Surrounding (MDS)

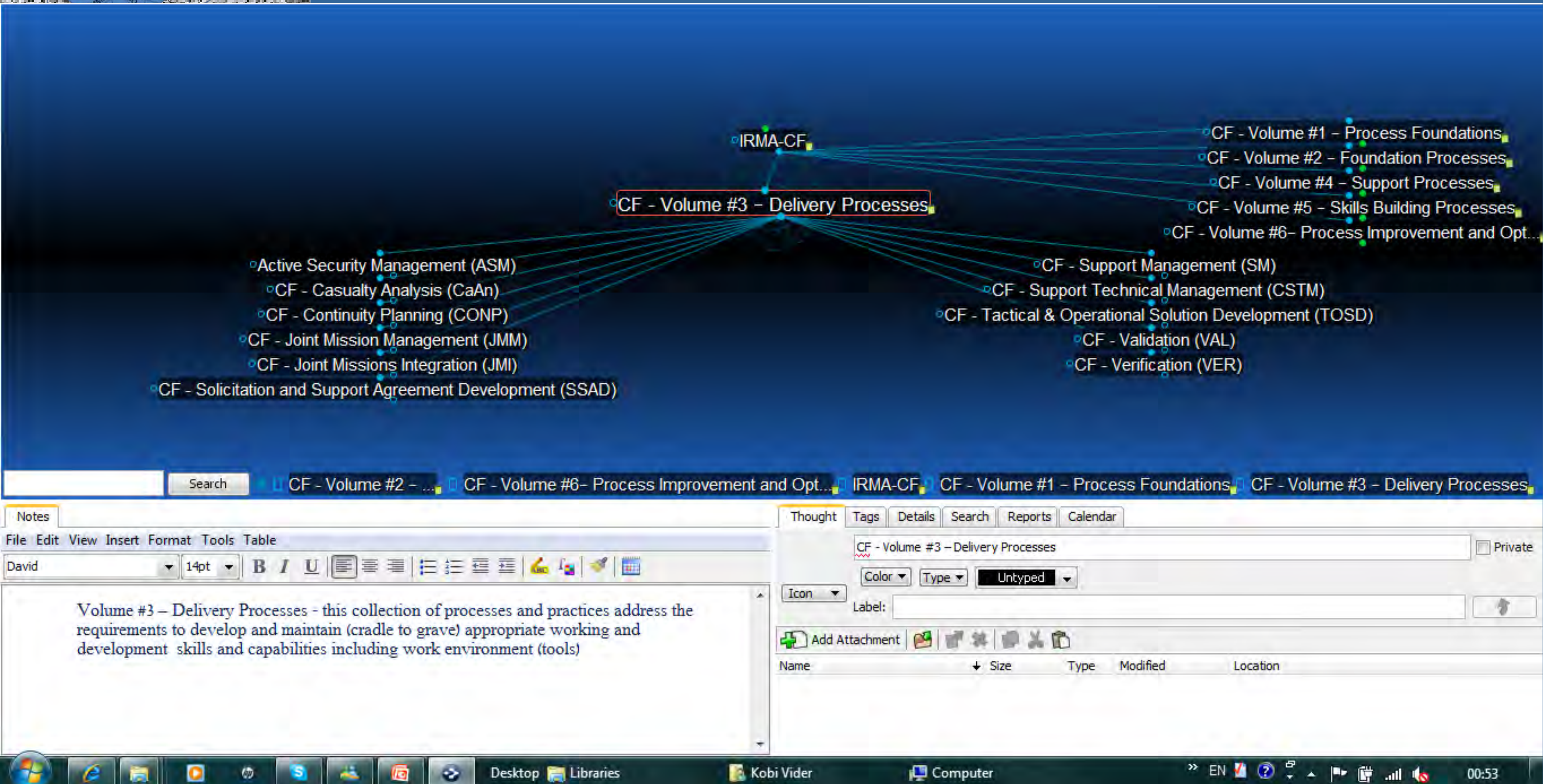
Model Processes

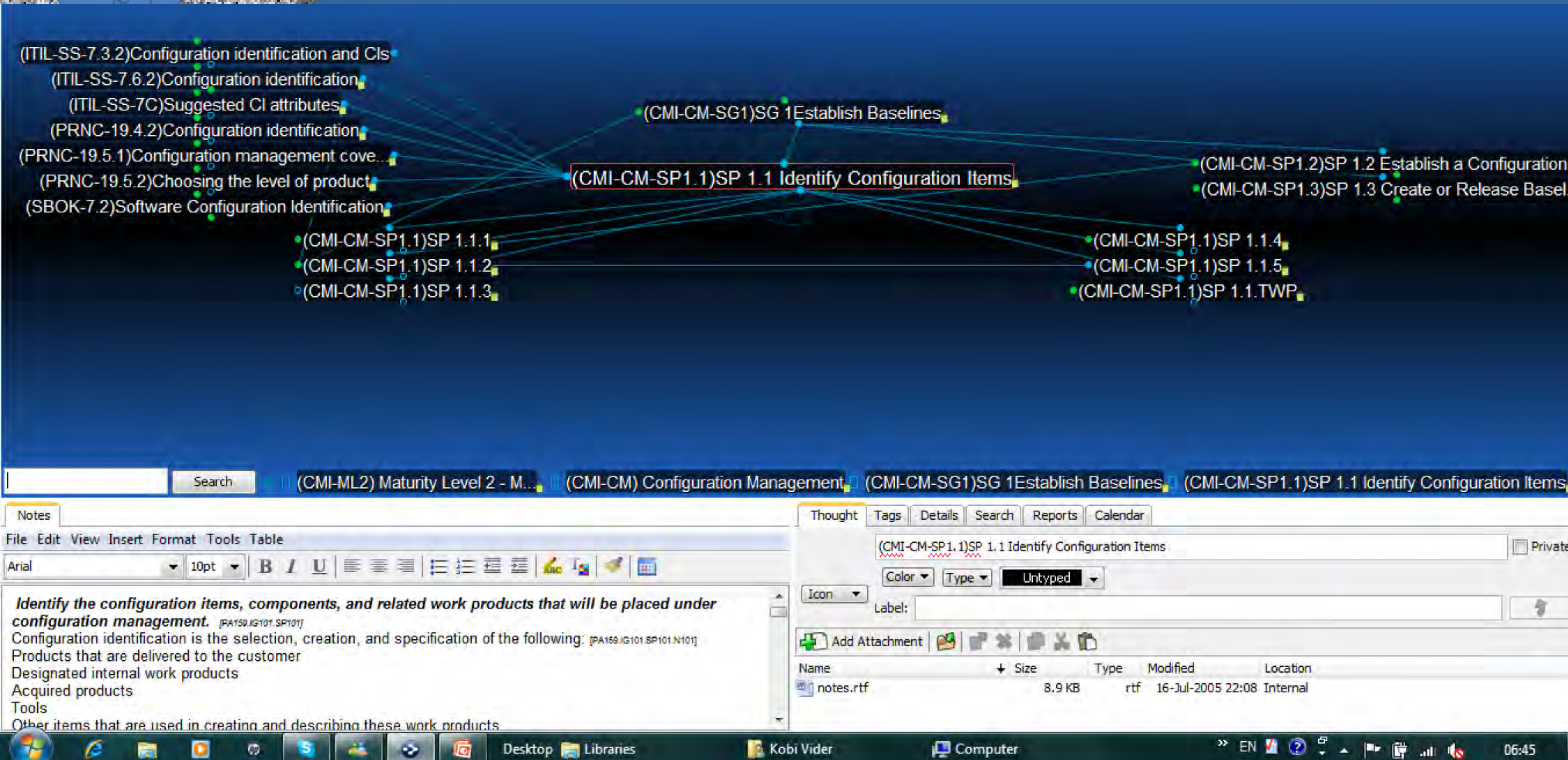
Volume 1 – Process Foundations	Volume 2 – Foundation Processes	Volume #3 – Delivery Processes	Volume #4 – Support Processes	Volumes#5 – Skills Building Processes	Volume #6– Process Improvement and Optimization Capabilities
1. Process Goals and Objectives 2. Process ingredients 3. Process Key Process Indicators (KPIs)	1. Business Objectives & Goals Management (BOGM) 2. Business Objectives & Goals Development (OGD) 3. Planning and Control 4. Business Measurement and Plan (BMP) 5. Business Scoping (BS) 6. Capacity and Availability Management (CAM) 7. Business Strategy Management (BSM)	1. Business Continuity (BCON) 2. Support Management (SM) 3. Support Technical Management (CSTM) 4. Solicitation and Support Agreement Development (SSAD) 5. Joint Mission Management (JMM) 6. Joint Missions Integration (JMI) 7. Tactical & Operational Solution Development (TOSD) 8. Validation (VAL) 9. Verification (VER)	1. Causal Analysis and Resolution (CAR) 2. Configuration Management (CM) 3. Risk Management (RSKM) 4. Incident Resolution and Prevention (IRP) 5. Service Delivery (SD) 6. Service System Development (SSD) 7. Service System Transition (SST)	1. Training (AUT) 2. Decision Analysis and Resolution (DAR)	1. Business and Operation Quality Assurance (BOQA) 2. Business Process Characterization (BPD) 3. Business Process Focus (BPF) 4. Business Unit Process Performance (BUPP) 5. Quantitative Business Management (QBM) 6. Business Innovation (BIn)

Detailed Examples and Elaborations

- Link to Model Map (Excel)
- Link to Model BOK (Word)
- Link to Model Scoping (Excel)
- Link to Model Checklist Chart (Visio)

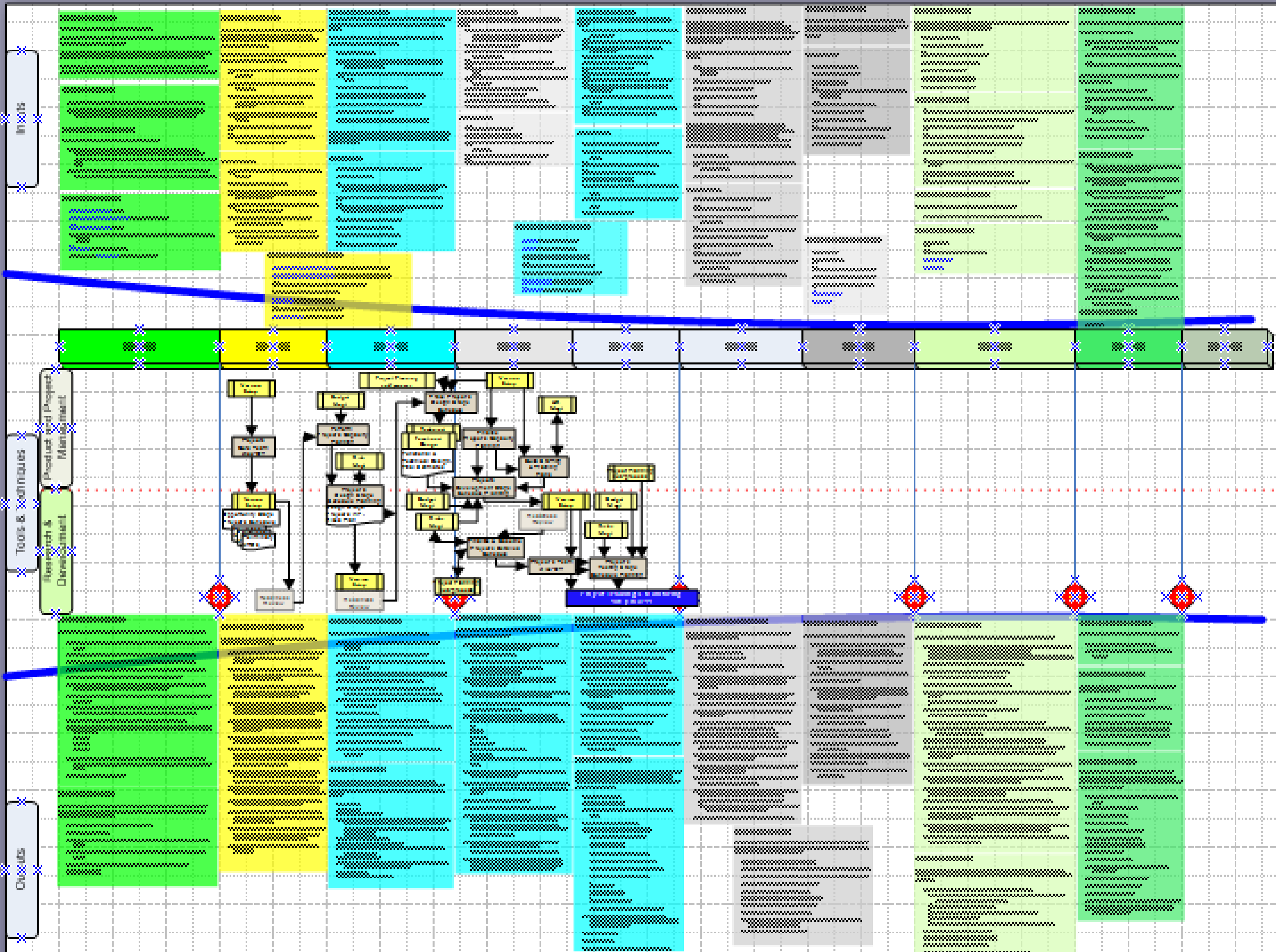

Performance Models
HERMLC
IRMA-AM
IRMA-B
IRMA-CFac
IRMA-OMR
IRMA-S
IRMA-CF
CF - Volume #1 – Process Foundations
CF - Volume #2 – Foundation Processes
CF - Volume #3 – Delivery Processes
CF - Volume #4 – Support Processes
CF - Volume #5 – Skills Building Processes
CF - Volume #6– Process Improvement and Optimization Capabilities
Search
Notes
File Edit View Insert Format Tools Table
David
14pt
IRMA-CF - Integrated Risk Management Approach Core Foundation, is the basic model that is the mandatory Body Of Knowledge (BOK) to all other models
Thought
Tags
Details
Search
Reports
Calendar
IRMA-CF
Private
Color
Type
Untyped
Icon
Label:
Add Attachment
Name
Size
Type
Modified
Location
Desktop
Libraries
Kobi Vider
Computer
EN
00:52


IRMA-CF
CF - Volume #1 – Process Foundations
CF - Volume #2 – Foundation Processes
CF - Volume #4 – Support Processes
CF - Volume #5 – Skills Building Processes
CF - Volume #6– Process Improvement and Opt...
CF - Volume #3 – Delivery Processes
Active Security Management (ASM)
CF - Casualty Analysis (CaAn)
CF - Continuity Planning (CONP)
CF - Joint Mission Management (JMM)
CF - Joint Missions Integration (JMI)
CF - Solicitation and Support Agreement Development (SSAD)
CF - Support Management (SM)
CF - Support Technical Management (CSTM)
CF - Tactical & Operational Solution Development (TOSD)
CF - Validation (VAL)
CF - Verification (VER)
Search
CF - Volume #2 - ...
CF - Volume #6– Process Improvement and Opt...
IRMA-CF
CF - Volume #1 – Process Foundations
CF - Volume #3 – Delivery Processes
Notes
File Edit View Insert Format Tools Table
David
14pt
Volume #3 – Delivery Processes - this collection of processes and practices address the requirements to develop and maintain (cradle to grave) appropriate working and development skills and capabilities including work environment (tools)
Thought Tags Details Search Reports Calendar
CF - Volume #3 – Delivery Processes
Private
Color Type Untyped
Icon
Label:
Add Attachment
Name Size Type Modified Location
Desktop Libraries
Kobi Vider
Computer
EN
00:53


(ITIL-SS-7.3.2)Configuration identification and CIs
(ITIL-SS-7.6.2)Configuration identification
(ITIL-SS-7C)Suggested CI attributes
(PRNC-19.4.2)Configuration identification
(PRNC-19.5.1)Configuration management cove...
(PRNC-19.5.2)Choosing the level of product
(SBOK-7.2)Software Configuration Identification
(CMI-CM-SG1)SG 1Establish Baselines
(CMI-CM-SP1.1)SP 1.1 Identify Configuration Items
(CMI-CM-SP1.2)SP 1.2 Establish a Configuration...
(CMI-CM-SP1.3)SP 1.3 Create or Release Basel...
(CMI-CM-SP1.1)SP 1.1.1
(CMI-CM-SP1.1)SP 1.1.2
(CMI-CM-SP1.1)SP 1.1.3
(CMI-CM-SP1.1)SP 1.1.4
(CMI-CM-SP1.1)SP 1.1.5
(CMI-CM-SP1.1)SP 1.1.TWP
Search
(CMI-ML2) Maturity Level 2 - M...
(CMI-CM) Configuration Management
(CMI-CM-SG1)SG 1Establish Baselines
(CMI-CM-SP1.1)SP 1.1 Identify Configuration Items
Notes
File Edit View Insert Format Tools Table
Arial 10pt
Identify the configuration items, components, and related work products that will be placed under configuration management. [PA159.IG101.SP101]
Configuration identification is the selection, creation, and specification of the following: [PA159.IG101.SP101.N101]
Products that are delivered to the customer
Designated internal work products
Acquired products
Tools
Other items that are used in creating and describing these work products
Thought Tags Details Search Reports Calendar
(CMI-CM-SP1.1)SP 1.1 Identify Configuration Items
Private
Color Type Untyped
Icon
Label:
Add Attachment
Name Size Type Modified Location
notes.rtf 8.9 KB rtf 16-Jul-2005 22:08 Internal
Desktop Libraries Kobi Vider Computer
EN
06:45

	Current Groups														New Functional Groups				
							R&D		System Analysis		Infrastructure								
Full Name	QA	IS	TW	PM (Delivery)	HD	PMO	DEV	PS	RA	SA	DBA	ALM	CM	IT	QA (Process)	M&A (& MM)	Process Group		
Causal Analysis and Resolution	2	2					1	1	1	1	1					3		12	1
Configuration Management													3					3	1
Decision Analysis and Resolution	2	2		3		1	2											10	1
Integrated Project Management	1	1		3		2	2		2								3	14	2
Integrated Project Management - IPPD																		0	0
Measurement and Analysis	1	2					1	1	1	1	1					3		11	1
Organizational Innovation and Deployment																		0	0
Organizational Process Definition												3					3	6	2
Organizational Process Definition - IPPD																		0	0
Organizational Process Focus															3		3	6	2
Organizational Process Performance																3		3	1
Organizational Training																		0	0
Product Integration							3					3						6	2
Project Monitoring and Control				3														3	1
Project Planning	3			3														6	2
Process and Product Quality Assurance															3			3	1
Quantitative Project Management	3															3		6	2
Requirements Development							3		3	3								9	3
Requirements Management							3		3	3								9	3
Risk Management				3														3	1
Supplier Agreement Management																		0	0
Technical Solution							3				3	3						9	3
Validation	3																	3	1
Verification	3						3											6	2
Capacity and Availability Management														3				3	1
Incident Resolution and Prevention		3																3	1
Service Continuity		3												3				6	2
Service Delivery		3																3	1
Service System Development		3																3	1
Service System Transition		3																3	1
Strategic Service Management		3																3	1
	18	25	0	15	0	3	21	2	10	8	5	9	3	6	6	12	9		
GP 2.1 Establish an Organizational Policy	3																	3	
GP 2.2 Plan the Process																		0	
GP 2.3 Provide Resources																		0	
GP 2.4 Assign Responsibility																		0	
GP 2.5 Train People	3																	3	
GP 2.6 Manage Configurations																		0	
GP 2.7 Identify and Involve Relevant Stakeholders																		0	
GP 2.8 Monitor and Control the Process																		0	
GP 2.9 Objectively Evaluate Adherence																		0	
GP 2.10 Review Status with Higher Level Management																		0	
	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
GP 3.1 Establish a Defined Process	3																	3	
GP 3.2 Collect Improvement Information	3																	3	
	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
GP 4.1 Establish Quantitative Objectives for the Process																		0	
GP 4.2 Stabilize Subprocess Performance	3																	3	
	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
GP 5.1 Ensure Continuous Process Improvement																		0	
GP 5.2 Correct Root Causes of Problems	3																	3	
	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
CL5	36	25	0	15	0	3	21	2	10	8	5	9	3	6	6	12	9		
CL4	33	25	0	15	0	3	21	2	10	8	5	9	3	6	6	12	9		
CL3	30	25	0	15	0	3	21	2	10	8	5	9	3	6	6	12	9		
CL2	24	25	0	15	0	3	21	2	10	8	5	9	3	6	6	12	9		

Sheet1 Sheet2 Sheet3


Inputs
Product and Project Management
Tools & Techniques
Research & Development
Outputs

The Model Sturdiness Capabilities Echelon

The Model Sturdiness Capabilities Echelon-1

- The Sturdiness Capabilities Echelon is used to describe an evolutionary progress for an organization that wants to improve its processes across the organization to develop and maintain its products and services.
- The model supports two progress or improvement paths:
  - Incessant - enabling an organization to incrementally improve processes corresponding to an individual functional group / specific domain area (or set of processes) selected by the organization / functional group
  - Predefined – the organization implements related predefined sets of processes

The Model Sturdiness Capabilities Echelon - 2

- These two improvement paths are associated with two types of echelon that correspond to the two views, Incessant and Predefined .
- For the Incessant view, we use the term Professionalism Group Capabilities Echelon – (GCE).
- For the staged representation, we use the term Organizational Sturdiness Echelon – (OSE).

The Model Sturdiness Capabilities Echelon - 3

- Regardless of the view you select, the concept of echelon is the same.
- Echelon characterize improvement from an ill-defined state to a state that uses quantitative information to determine and manage improvements that are needed to meet an organization’s business objectives.
- To reach a particular echelon, an organization must satisfy all of the appropriate model entities or set of processes that are targeted for improvement, regardless of what the volume or selection of domains. (refer to the scoping map)

The Model Sturdiness Capabilities Echelon - 4

- A capability echelon consists of a process foundations and its related ingredients that can improve the organization’s processes associated.
- Capability echelons provide a scale for measuring your processes against each process area in the model.
- Each echelon is a layer in the foundation for continuous process improvement.
- Capability echelons are cumulative (i.e., a higher echelon includes the ingredients of the lower levels).

Statistically Managing Your Processes - 1

- Determine whether processes are behaving consistently or have stable trends (i.e., are predictable)
- Identify processes where the performance is within natural bounds that are consistent across process implementation teams
- Establish criteria for identifying whether a process or process element should be statistically managed, and determine the pertinent measures and analytic techniques to be used in such management
- Identify processes that show unusual (e.g., sporadic or unpredictable) behavior
- Identify any aspects of the processes that can be improved in the organization's set of standard processes
- Identify the implementation of a process which performs best

Statistically Managing Your Processes - 2

- Root Cause Analysis & Resolution
  - Identify and analyze causes of defects and other problems
  - Take specific actions to remove the causes

- The ‘project’ can then take actions to prevent the occurrence of those types of defects and problems in the future
- Many ‘projects’ implement it to identify and eliminate special cause variations to stabilize the process

Suggested KPI's to Measure Process Success

- Operability Predictability
- Response Time Predictability
- Cost of Rectifying Problems
- Survivability Predictability
- Productivity
- Total Cost of Risk
- Recovery (to L'0') time
- Supply Chain Response Time
- Response Efficiency
- Operability Continuity
- Survivability Continuity

Operational Processes KPI’s

- Known Capability and Stable
- Defined Ingredients
- Known Critical Elements
- Meeting Objectives
- Controlled Interfaces
- Responsive / Modifiable
- Resilience / “Agile”
- Relevant ‘What If’s Scenarios
- Accepted Tolerance / Freedom Boundaries
- Predictable Outcomes

- Influence of Critical Elements on process output
- Process resources utilization ‘What If’s Scenarios
- Process elements capability
- Quantitative definition of process ingredients

System Compliances' KPI’s

- Scalability
- Availability
- Reliability
- Serviceability
- Maintainability
- Supportability
- Stability
- Reusability
- Soundness of Technology Future

- Technology flexibility
- Capacity growth models
- System (size) growth models
- Time to Restore
- Down time
- MTBF
- Support calls causes and density
- Technology extendibility

HERMES
Applying Evaluation and Assessments to the STORM

HERMES

- Standard Description Document (SDD)
- Mandatory Evaluation Plan (MEP) with tailoring guidelines and preconfigured sets to address the five models
- Interpretation Guidelines Sets (IGS) addressing the five models
- Detailed scoping and rating scheme

Link to Folder

Link to SDD

What We Look For In Appraisals - 1

- Indicators of:
  - Culture
  - Dependencies
  - Critical issues that effect the operational concept
- Planning approaches for complex / matrix environments
- Inter-unit coordination throughout the processes
- External coordination throughout processes
- Considerations of development of inter protocols or best practices
- Inter-organizational communication as an integral ingredient in the operational environment

What We Look For In Appraisals - 2

- Relationships
- Authority
- Strategic vs. operational vs. tactical
- Coordination
- Direction

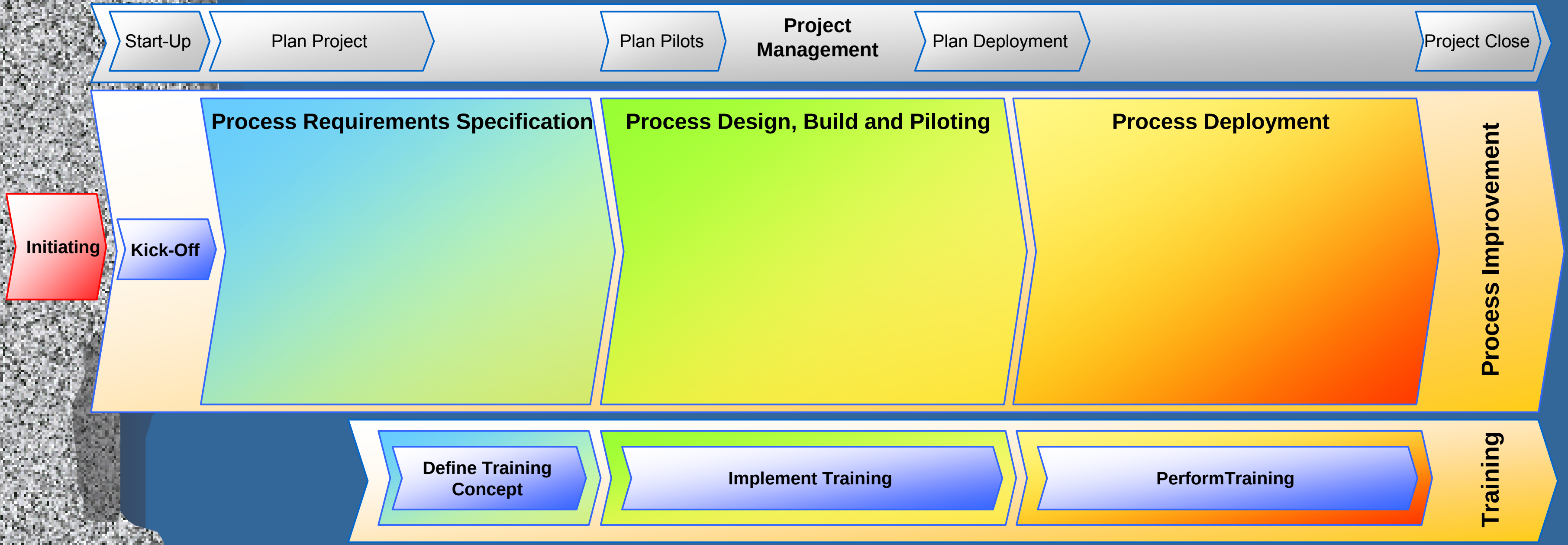
Implementation Journey Guidelines

- Awareness and Orientation Workshop
- Organizational Mapping, Scoping the Specific Needs
- Developing Measurable Objectives
- Developing and Presenting an Organizational Related Case Study
- Gap Analysis Planning
- Performing the Gap Analysis
- Developing and Presenting the Improvement Plan
- Implementation Phase and Ongoing Progress Checks
- Evaluation
- Ongoing Activities

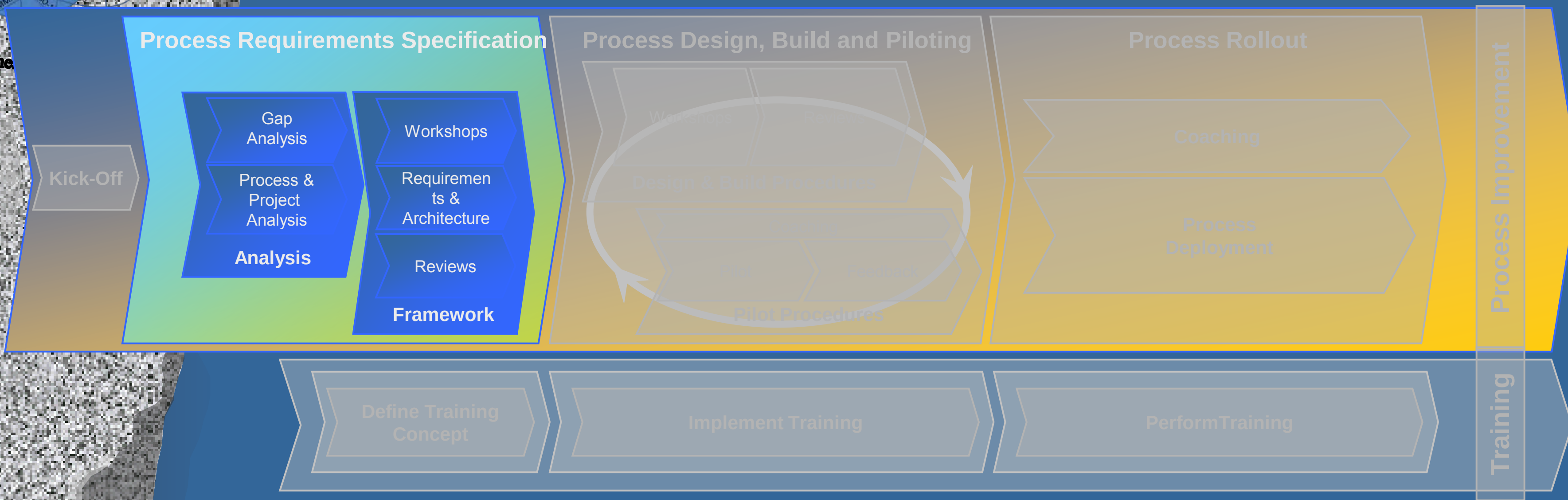
How it’s done

Short discussion

Overall Project

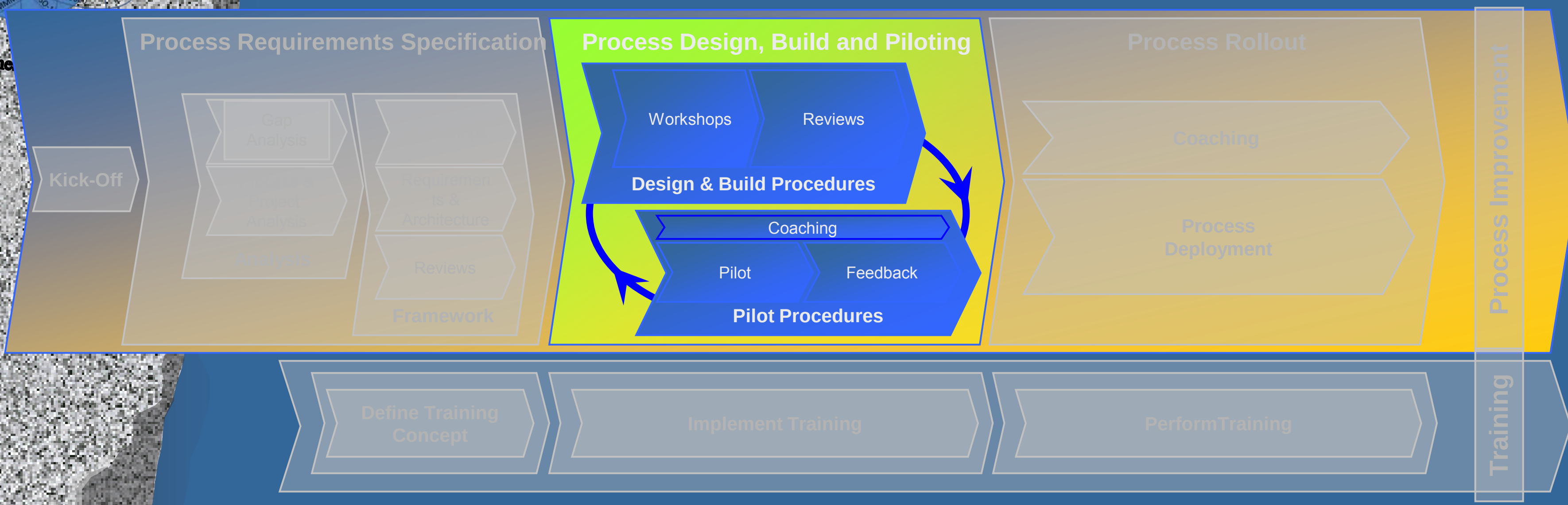


Process Requirements Specification



- Analysis
  - Informal gap analysis / Post Mortem
  - Basis for improvement planning
  - Result: report of assessment / gap analysis with improvement suggestions

Process Design, Build and Piloting



- Definition of usable processes “ready for life”
- Methods
  - Workshops for definition processes
  - Reviews (workshops / offline)
  - Coaching and piloting
  - Collecting feedback from pilot projects (e.g. interviews/workshops)
- Result: defined process (descriptions, templates, examples, …)

K.V.P Consulting

Multi Disciplines Surrounding (MDS)

Organizational Processes and Lifecycles

Models References

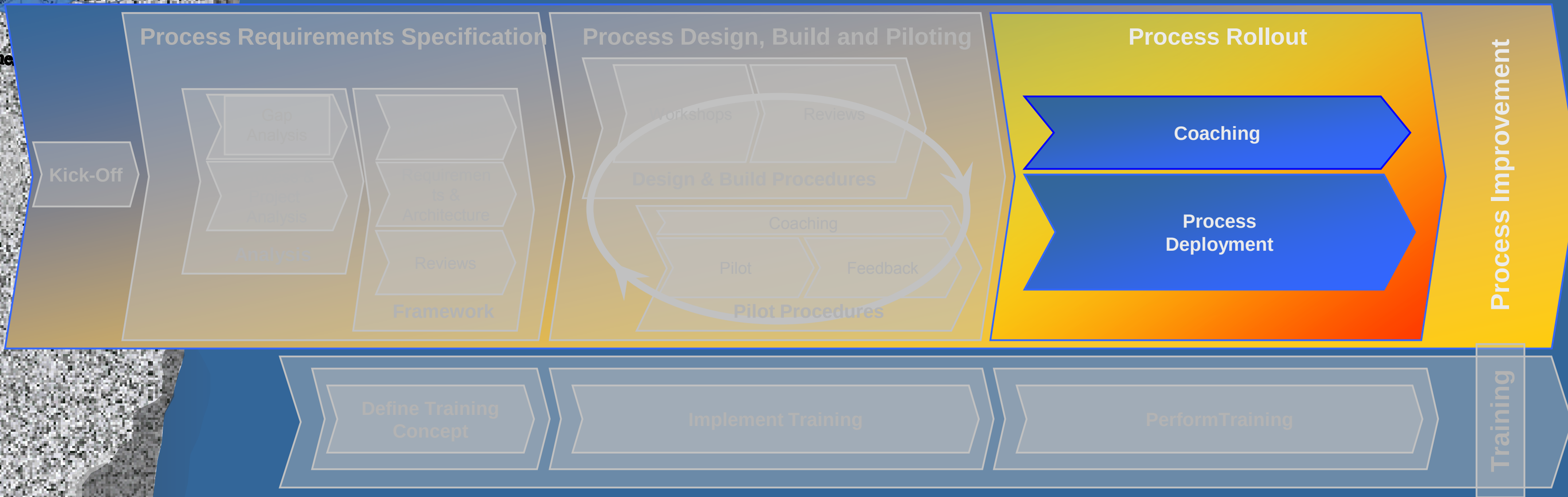
Compliance Mapping

Best Practices and Processes

Measurements Library

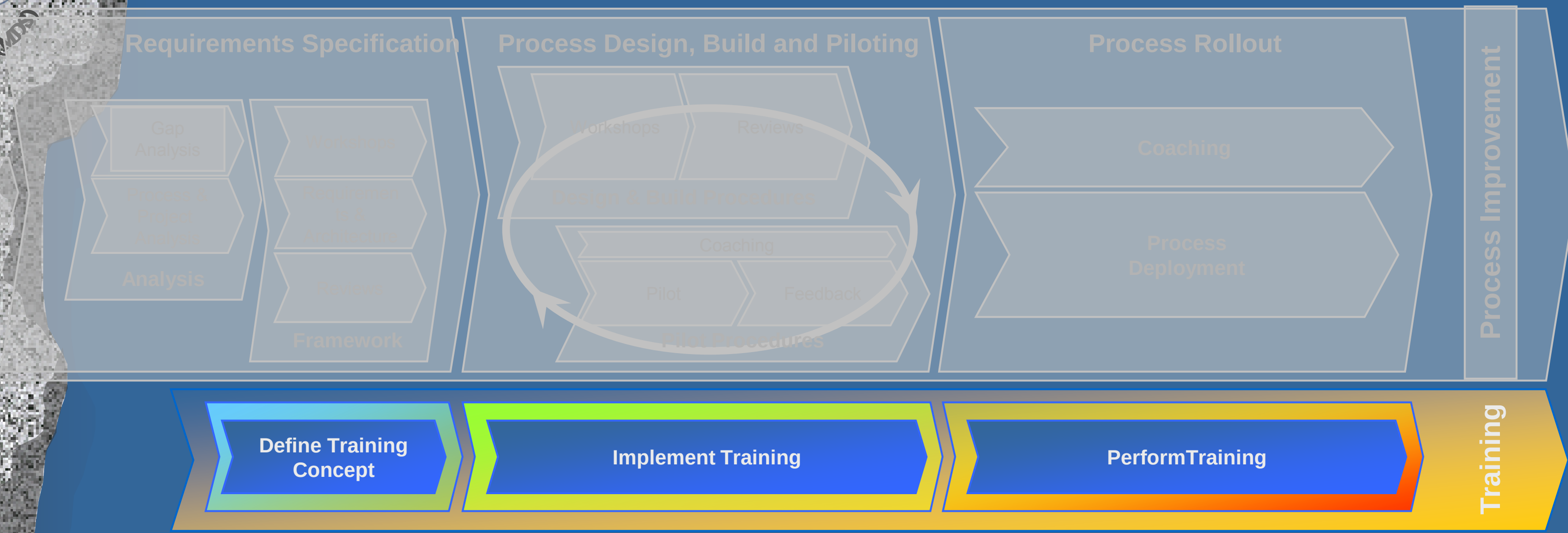
Statistical Readiness

Process Rollout



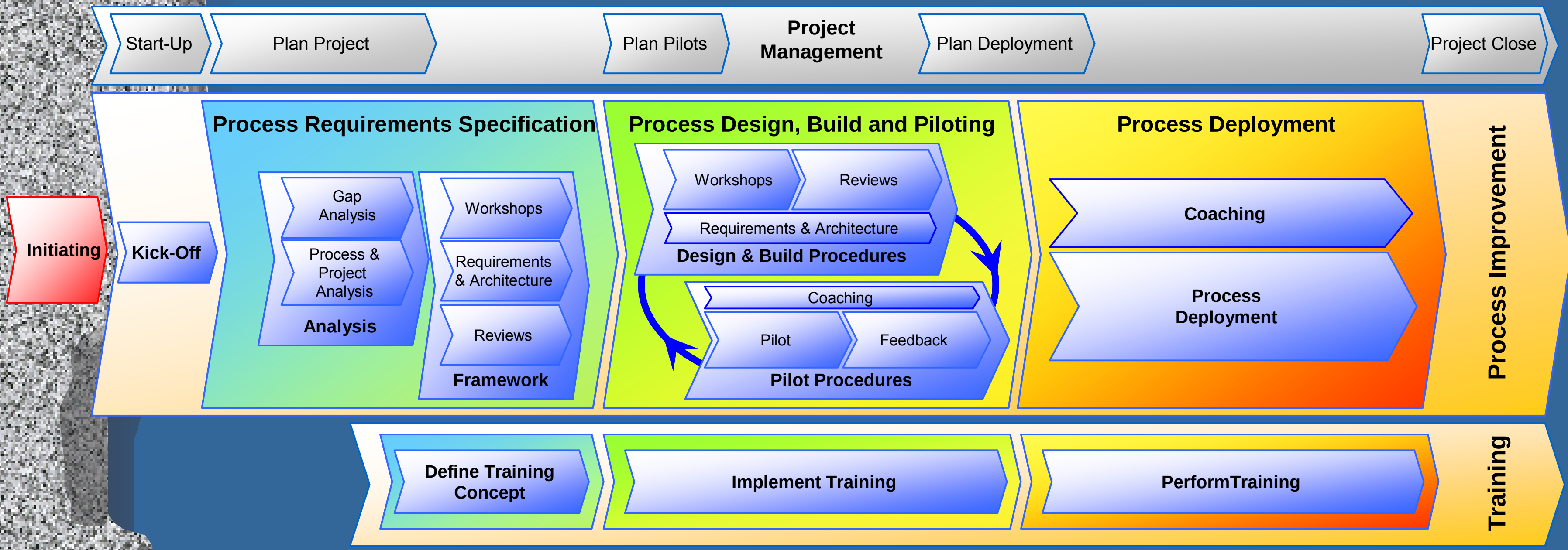
- Process Rollout
  - Processes are used in (new) current units
  - Training and coaching of project members
  - Collection and evaluation of measurements
  - Collection of feedback for following improvement cycles
  - Result:
    deployed process, initial measurements and improvement suggestions

Training



- Identify roles to be trained
- Schedule of the training (project / role specific)
- Contents: processes / tools / methods to be trained
- Creation of exercises
- Performance of trainings

Overall Proceeding



STORM (Strategic Technology and Operational Risk Management)

Innovative Approach for Organizational Integrated Risk Management Approach

Kobi Vider – Picker

K.V.P Consulting

Kobi.Vider@hotmail.com

+972522946676

Intellectual Property Disclaimer

- The content of this presentation, STORM / IRMA / HERMES models and all related materials are the intellectual property of Kobi Vider and K.V.P Consulting
- THE CONTENT ON THIS WEBSITE IS PROVIDED "AS IS" WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND EITHER EXPRESSED OR IMPLIED. TO THE FULLEST EXTENT POSSIBLE PURSUANT TO APPLICABLE LAW, Kobi Vider and K.V.P Consulting DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, IMPLIED WARRANTIES OR CONDITIONS OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE,
- All materials contained in this presentation and all models related documentation are protected by copyright laws, and may not be reproduced, republished, distributed, transmitted, displayed, broadcast or otherwise exploited in any manner without the express prior written permission of Kobi Vider and K.V.P Consulting
- Without a written approval from Kobi Vider or K.V.P Consulting; you are not permitted to modify the Content. You are not permitted to make use of the Content for commercial gain. You must keep intact all copyright notices for the Content and provide, reasonable to the medium or means you are utilizing, the name of Access Copyright and the title of the Content if supplied. Such credit may be provided in any reasonable manner

Commercial Port STORM Pilot

Case Study

Background to the Need

- Critical facility emergency events and incidents are managerial, not technical
- Mission and objective statement as much as other, must include quantitative objectives that are stated in a clear way
- Basic building block is the capability to accurately evaluate the unit's effectiveness along with the efficiency of its resource usage
- The main challenge is to integrated the overall **risks** in the '**spider net**' and to **understand** their **true impact**

STORM Gap Analysis Main Activities

1. Identifying critical components of information needs and knowledge gaps their origins
2. Identification, mapping and analysis of critical components (units, facilities, infrastructure, people)
3. Threats identification and analysis
4. Identification, mapping and analysis of sensitive areas and points, weak points and related damage / impact to objectives
5. Risk identification, mapping and analysis, respectively to the threats
6. Risk management and measurements

Conceptual Case Study


Image © 2010 DigitalGlobe
©2009 Google

Port - Background

Business Objectives

- Port of Civitavecchia is a busy ferry port located 80 km / 50 miles west north west of Rome and providing both
  - Passenger and
  - Cargo
    services to
  - Italian and
  - European
    destinations
- The ferry terminal offers an impressive selection of passenger amenities which include
  - ATMs
  - Information bureaux
  - Waiting rooms
  - Left luggage facilities and
  - Cafeterias

Port - Background

Reference Threats (for this presentation only)

- Passengers
  - Personal safety
  - Public safety
  - Luggage loss and damages
  - Public security (civilian and crime)
- Cargo
  - Loss and damages
  - Misshipment
  - Thefts
  - Smuggling
  - Storage
  - Management (special needs) and maintenance
- Italian (Local)
  - Uncontrolled movements
- European (Export)
  - Regulations
  - Illegal immigration

Passengers


Crystal Serenity Bridgecam Sat Apr 26 13:07:32 2008
Position: Docked in Rome/Civitavecchia, Italy

Cargo


Cargofresh
©2009 Google

Port - Background

Reference Threats (for this presentation only)

- ATMs
  - Frauds
  - Pickpocketing
  - Identity thefts
- Information bureaux,
  - Fraud chain
  - Illegal services / Activity
  - Satellite unapproved services / Activity
- Waiting rooms
  - Pickpocketing
  - Luggage thefts
  - Public order
- Left luggage facilities
  - Frauds
  - Luggage thefts
  - Smuggling and fraud chain
- Cafeterias
  - Food Quality
  - Food Safety
  - Illegal services / Activity
  - Pickpocketing
  - Thefts
  - Frauds

Port - Background

- Applicable STORM (IRMA) model and Components
  - IRMA-B Selected Components
  - IRMA-CF Selected Components
  - IRMA-AM Selected Components
  - IRMA-OMR Selected Components
- HERMES

Port - Background

- Analysis approach and method
  - Visual Screening
  - Hidden observation and simulation
  - Process simulation (tool based)
- Main Risks **(partial list for this presentation only)**
  - Leading
    - Physical Casualties
    - Material damages
    - Availability level
    - Operational continuity

- Consequenced
  - Branding
  - Perception
  - Revenue
  - Position

Port - Background

- Main Measurements **(partial list for this presentation only)**
  - Physical Casualties
    - Severity
    - Density vs. causes
  - Material damages
    - The human cost of the security system / calculated against the cost of damage
  - Availability level
    - Unavailability time vs. cost
    - Unavailability time vs. perception
  - Operational continuity
    - Mean time between failures
    - Time to recovery
    - Recovery levels (the just good enough)
    - The cost of inspection and assessment of continuity components against the expected damage

Port - Background

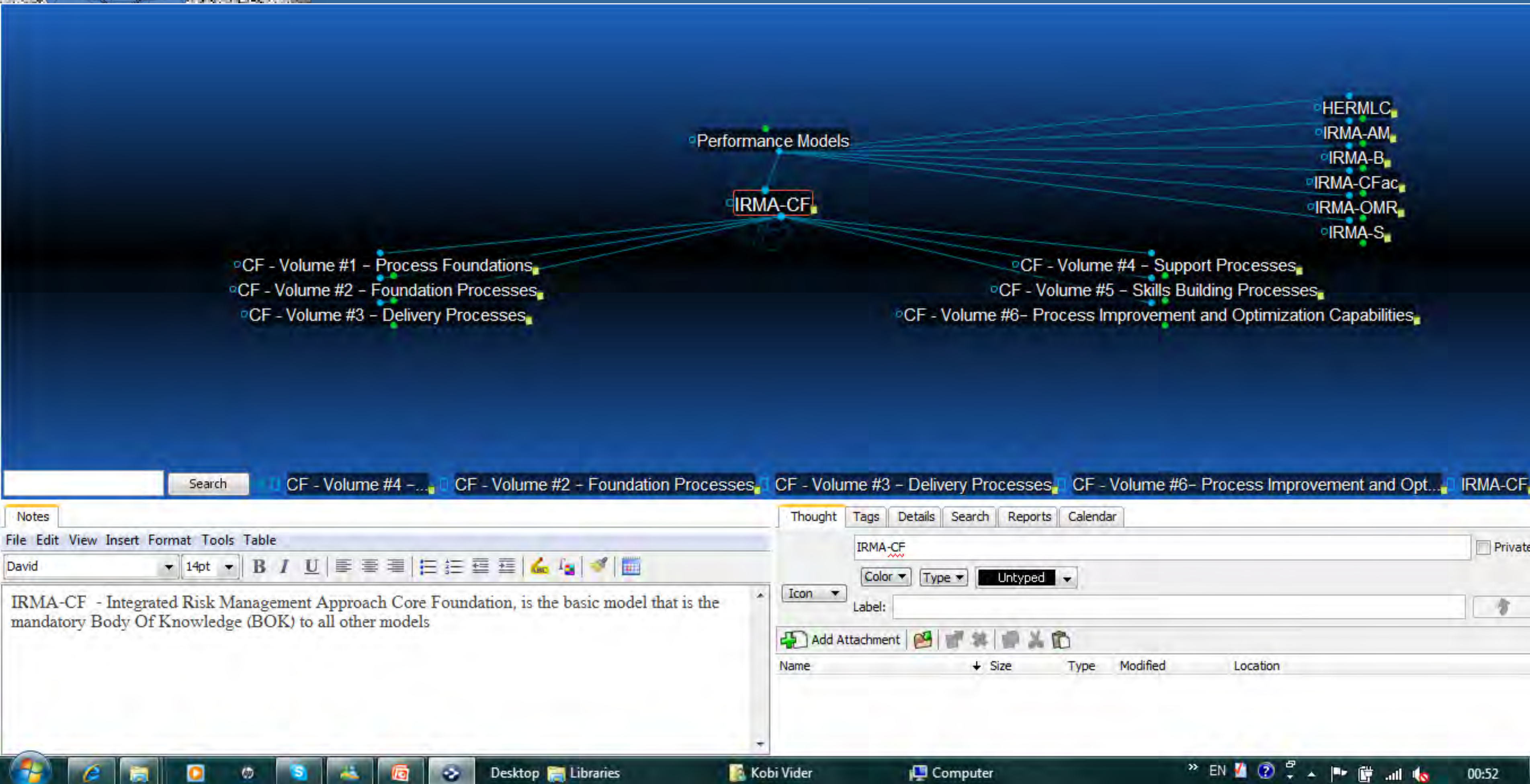
- Main Measurements **(partial list for this presentation only)**
  - Branding
    - Benchmarks
  - Perception
    - Customer satisfaction
  - Revenue
    - Cost and quality assurance activities
    - Cost op poor quality
  - Position
    - Passengers trending

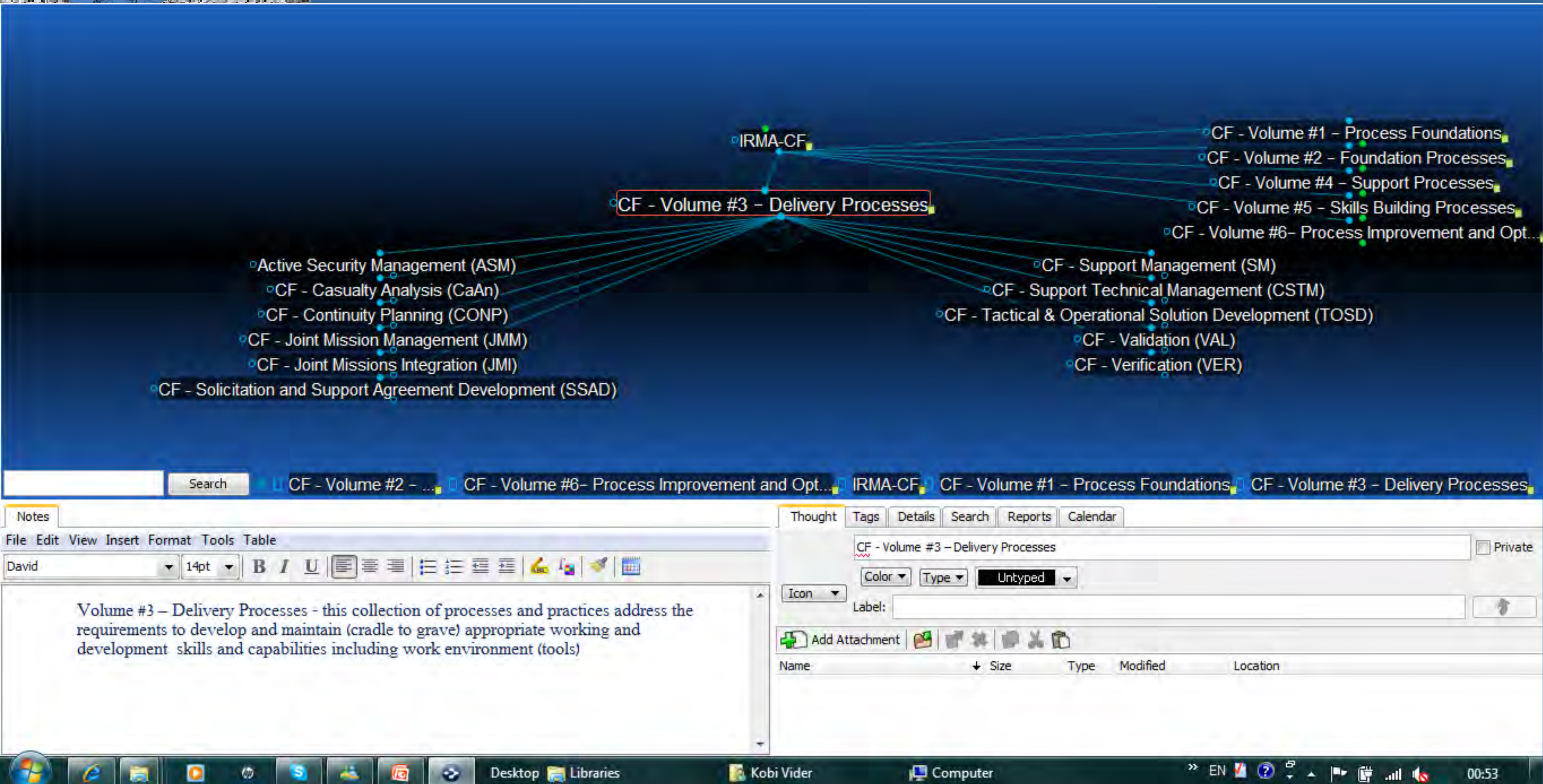
K.V.P Consulting

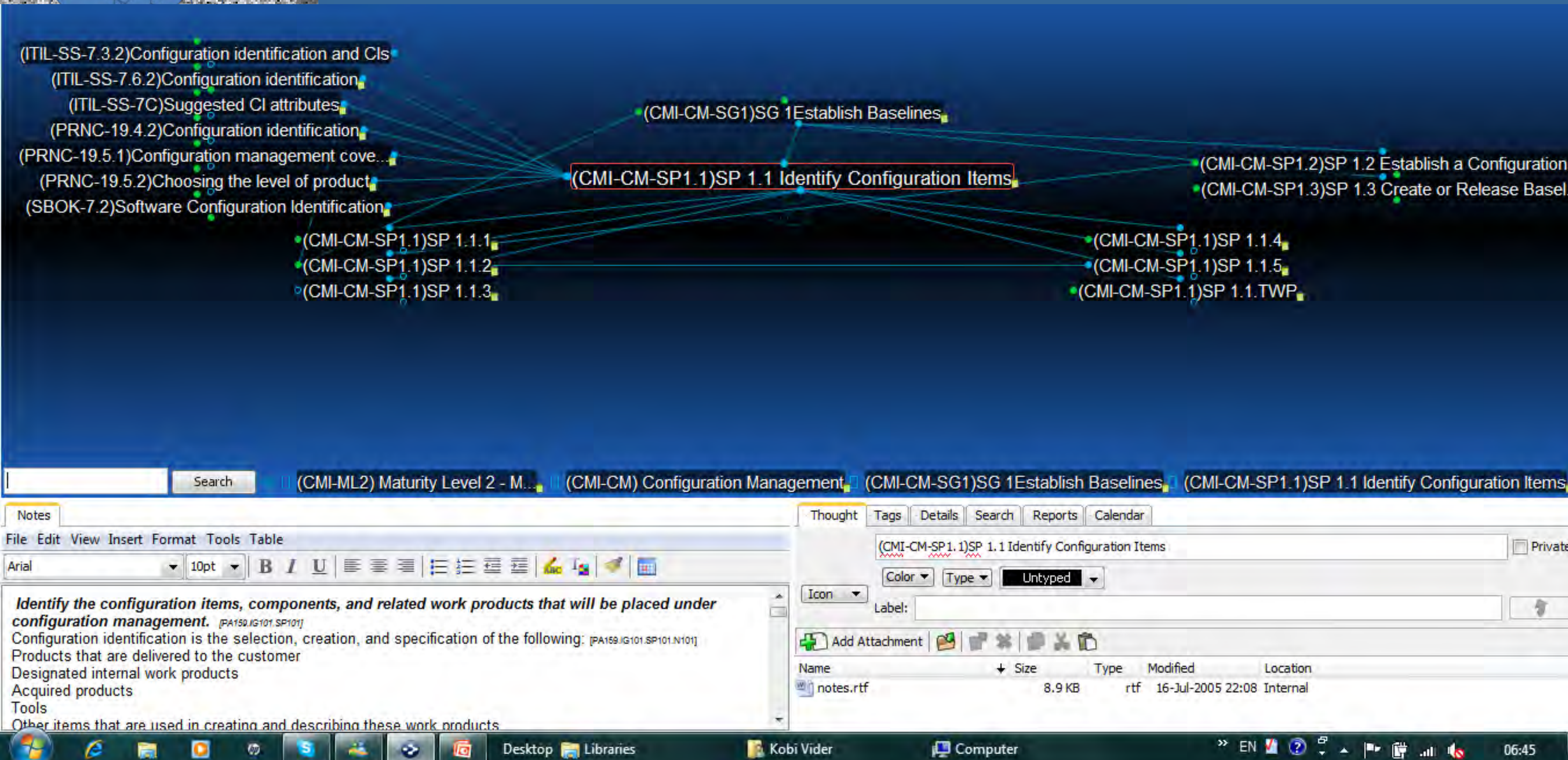
Multi Disciplines Surrounding (MDS)

Detailed Examples and Elaborations

- Link to Model Map (Excel)
- Link to Model BOK (Word)
- Link to Model Scoping (Excel)
- Link to Model Checklist Chart (Visio)

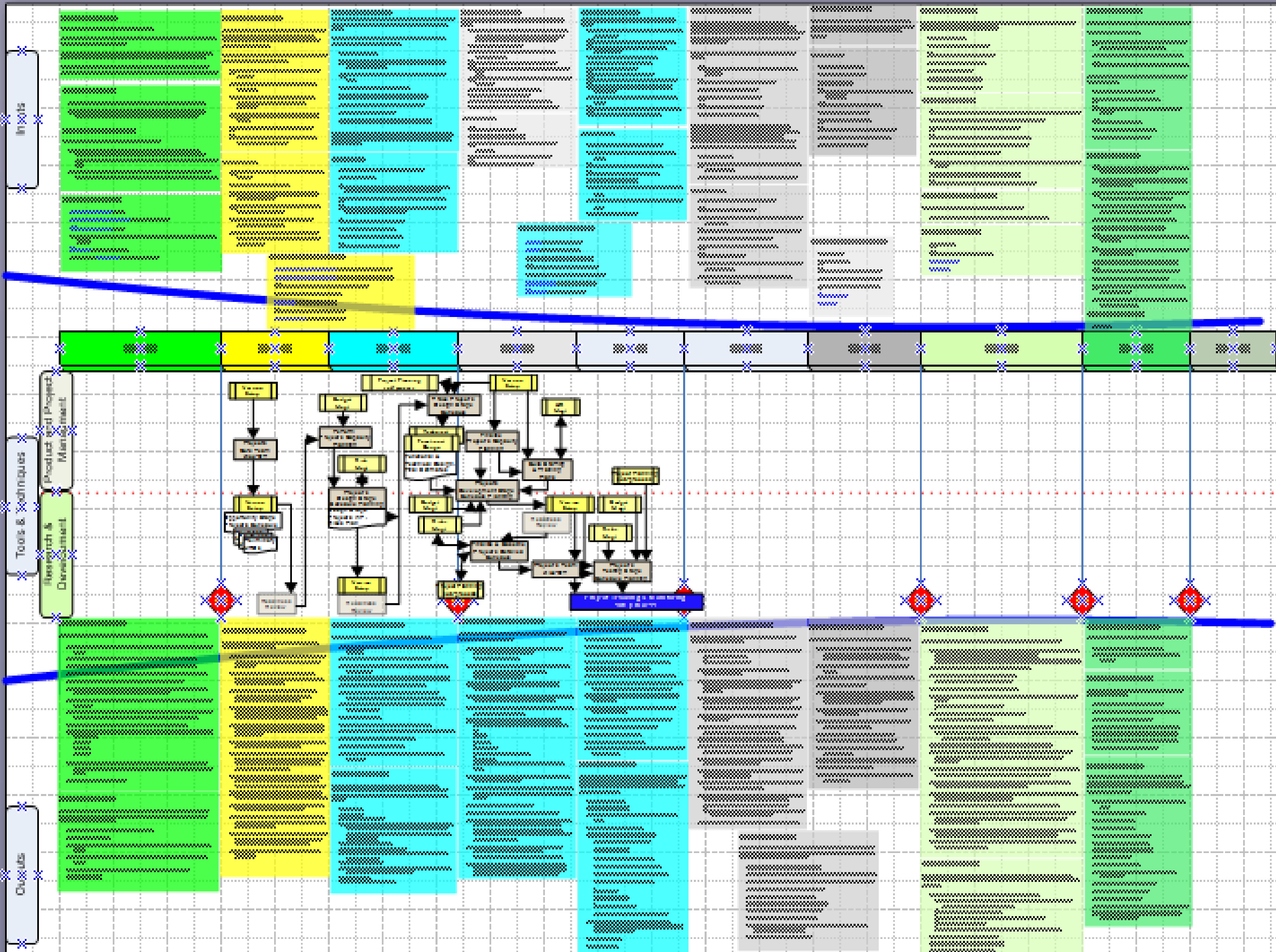

Performance Models
HERMLC
IRMA-AM
IRMA-B
IRMA-CFac
IRMA-OMR
IRMA-S
IRMA-CF
CF - Volume #1 – Process Foundations
CF - Volume #2 – Foundation Processes
CF - Volume #3 – Delivery Processes
CF - Volume #4 – Support Processes
CF - Volume #5 – Skills Building Processes
CF - Volume #6– Process Improvement and Optimization Capabilities
Search
CF - Volume #4 –...
CF - Volume #2 – Foundation Processes
CF - Volume #3 – Delivery Processes
CF - Volume #6– Process Improvement and Opt...
IRMA-CF
Notes
File Edit View Insert Format Tools Table
David
14pt
IRMA-CF - Integrated Risk Management Approach Core Foundation, is the basic model that is the mandatory Body Of Knowledge (BOK) to all other models
Thought Tags Details Search Reports Calendar
IRMA-CF
Private
Color Type Untyped
Icon
Label:
Add Attachment
Name Size Type Modified Location
Desktop Libraries
Kobi Vider
Computer
EN
00:52


IRMA-CF
CF - Volume #3 – Delivery Processes
CF - Volume #1 – Process Foundations
CF - Volume #2 – Foundation Processes
CF - Volume #4 – Support Processes
CF - Volume #5 – Skills Building Processes
CF - Volume #6– Process Improvement and Opt...
Active Security Management (ASM)
CF - Casualty Analysis (CaAn)
CF - Continuity Planning (CONP)
CF - Joint Mission Management (JMM)
CF - Joint Missions Integration (JMI)
CF - Solicitation and Support Agreement Development (SSAD)
CF - Support Management (SM)
CF - Support Technical Management (CSTM)
CF - Tactical & Operational Solution Development (TOSD)
CF - Validation (VAL)
CF - Verification (VER)
Search
CF - Volume #2 - ...
CF - Volume #6– Process Improvement and Opt...
IRMA-CF
CF - Volume #1 – Process Foundations
CF - Volume #3 – Delivery Processes
Notes
File Edit View Insert Format Tools Table
David
14pt
Volume #3 – Delivery Processes - this collection of processes and practices address the requirements to develop and maintain (cradle to grave) appropriate working and development skills and capabilities including work environment (tools)
Thought Tags Details Search Reports Calendar
CF - Volume #3 – Delivery Processes
Private
Color
Type
Untyped
Icon
Label:
Add Attachment
Name
Size
Type
Modified
Location
Desktop
Libraries
Kobi Vider
Computer
EN
00:53


(ITIL-SS-7.3.2)Configuration identification and CIs
(ITIL-SS-7.6.2)Configuration identification
(ITIL-SS-7C)Suggested CI attributes
(PRNC-19.4.2)Configuration identification
(PRNC-19.5.1)Configuration management cove...
(PRNC-19.5.2)Choosing the level of product
(SBOK-7.2)Software Configuration Identification
(CMI-CM-SG1)SG 1Establish Baselines
(CMI-CM-SP1.1)SP 1.1 Identify Configuration Items
(CMI-CM-SP1.2)SP 1.2 Establish a Configuration...
(CMI-CM-SP1.3)SP 1.3 Create or Release Basel...
(CMI-CM-SP1.1)SP 1.1.1
(CMI-CM-SP1.1)SP 1.1.2
(CMI-CM-SP1.1)SP 1.1.3
(CMI-CM-SP1.1)SP 1.1.4
(CMI-CM-SP1.1)SP 1.1.5
(CMI-CM-SP1.1)SP 1.1.TWP
Search
(CMI-ML2) Maturity Level 2 - M...
(CMI-CM) Configuration Management
(CMI-CM-SG1)SG 1Establish Baselines
(CMI-CM-SP1.1)SP 1.1 Identify Configuration Items
Notes
File Edit View Insert Format Tools Table
Arial 10pt
Identify the configuration items, components, and related work products that will be placed under configuration management. [PA159.IG101.SP101]
Configuration identification is the selection, creation, and specification of the following: [PA159.IG101.SP101.N101]
Products that are delivered to the customer
Designated internal work products
Acquired products
Tools
Other items that are used in creating and describing these work products
Thought Tags Details Search Reports Calendar
(CMI-CM-SP1.1)SP 1.1 Identify Configuration Items
Private
Color Type Untyped
Icon
Label:
Add Attachment
Name Size Type Modified Location
notes.rtf 8.9 KB rtf 16-Jul-2005 22:08 Internal
Desktop Libraries Kobi Vider Computer
EN 06:45

	Current Groups														New Functional Groups				
							R&D		System Analysis		Infrastructure								
Full Name	QA	IS	TW	PM (Delivery)	HD	PMO	DEV	PS	RA	SA	DBA	ALM	CM	IT	QA (Process)	M&A (& MMI)	Process Group		
Causal Analysis and Resolution	2	2					1	1	1	1	1					3		12	1
Configuration Management													3					3	1
Decision Analysis and Resolution	2	2		3		1	2											10	1
Integrated Project Management	1	1		3		2	2		2								3	14	2
Integrated Project Management +IPPD																		0	0
Measurement and Analysis	1	2					1	1	1	1	1					3		11	1
Organizational Innovation and Deployment																		0	0
Organizational Process Definition												3					3	6	2
Organizational Process Definition +IPPD																		0	0
Organizational Process Focus															3		3	6	2
Organizational Process Performance																3		3	1
Organizational Training																		0	0
Product Integration							3					3						6	2
Project Monitoring and Control				3														3	1
Project Planning	3			3														6	2
Process and Product Quality Assurance															3			3	1
Quantitative Project Management	3															3		6	2
Requirements Development							3		3	3								9	3
Requirements Management							3		3	3								9	3
Risk Management				3														3	1
Supplier Agreement Management																		0	0
Technical Solution							3				3	3						9	3
Validation	3																	3	1
Verification	3						3											6	2
Capacity and Availability Management														3				3	1
Incident Resolution and Prevention		3																3	1
Service Continuity		3												3				6	2
Service Delivery		3																3	1
Service System Development		3																3	1
Service System Transition		3																3	1
Strategic Service Management		3																3	1
	18	25	0	15	0	3	21	2	10	8	5	9	3	6	6	12	9		
GP 2.1 Establish an Organizational Policy	3																	3	
GP 2.2 Plan the Process																		0	
GP 2.3 Provide Resources																		0	
GP 2.4 Assign Responsibility																		0	
GP 2.5 Train People	3																	3	
GP 2.6 Manage Configurations																		0	
GP 2.7 Identify and Involve Relevant Stakeholders																		0	
GP 2.8 Monitor and Control the Process																		0	
GP 2.9 Objectively Evaluate Adherence																		0	
GP 2.10 Review Status with Higher Level Management																		0	
	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
GP 3.1 Establish a Defined Process	3																	3	
GP 3.2 Collect Improvement Information	3																	3	
	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
GP 4.1 Establish Quantitative Objectives for the Process																		0	
GP 4.2 Stabilize Subprocess Performance	3																	3	
	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
GP 5.1 Ensure Continuous Process Improvement																		0	
GP 5.2 Correct Root Causes of Problems	3																	3	
	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
CL5	36	25	0	15	0	3	21	2	10	8	5	9	3	6	6	12	9		
CL4	33	25	0	15	0	3	21	2	10	8	5	9	3	6	6	12	9		
CL3	30	25	0	15	0	3	21	2	10	8	5	9	3	6	6	12	9		
CL2	24	25	0	15	0	3	21	2	10	8	5	9	3	6	6	12	9		

Sheet1 Sheet2 Sheet3


Inputs
Tools & Techniques
Product and Project Management
Research & Development
Outputs

Tools Box Example

- Risk Evaluation Checklist
- Facility Management File
- DRP TOC
- BCP TOC
- Decision Tree Template
- Dynamic Knowledge Tree and Map

Risk Evaluation Checklist

Business Continuity Plan (BCP)

Complete Audit Checklist

No	Procedures	Status	Notes
1	Determine examination scope and objectives for reviewing the Business Continuity Plan (BCP) program.		
2	Determine the existence of an appropriate enterprisewide Business Continuity Plan (BCP).		
3	Determine the quality of Business Continuity Plan (BCP) oversight and support provided by the board of directors and senior management.		
4	Determine whether an adequate Business Impact Analysis (BIA) and risk assessment have been completed.		
5	Determine whether appropriate risk management over the Business Continuity Plan (BCP) process is in place.		
6	Determine whether the Business Continuity Plan (BCP) include appropriate testing to ensure the business process will be maintained, resumed, and/or recovered as intended.		
7	Determine whether the IT environment has a properly documented Business Continuity plan that complements the enterprise-wide and other departmental Business Continuity plans.		
8	Determine whether the Business Continuity Plan (BCP) include appropriate hardware backup and recovery.		
9	Determine whether the Business Continuity process includes appropriate data and application software backup and recovery.		
10	Determine whether the Business Continuity Plan (BCP) include appropriate preparation to ensure the data center recovery processes will work as intended.		
11	Determine whether the Business Continuity Plan (BCP) include appropriate security procedures.		
12	Determine whether the Business Continuity Plan (BCP) address critical outsourced activities.		
13	Discuss corrective action and communicate		

Data RecoveryTemplates and Checklist

Conducting a recovery test

No	Activity	Status Y	Status N	Status N/A	Notes
1	Select the purpose of the test. What aspects of the plan are being evaluated?				
2	Describe the objectives of the test. How will you measure successful achievement of the objectives?				
3	Meet with management and explain the test and objectives. Gain their agreement and support.				
4	Have management announce the test and the expected completion time.				
5	Collect test results at the end of the test period.				
6	Evaluate results. Was recovery successful? Why or why not?				
7	Determine the implications of the test results. Does successful recovery in a simple case imply successful recovery for all critical jobs in the tolerable outage period?				
8	Make recommendations for changes. Call for responses by a given				
9	Notify other areas of results. Include users and auditors.				
10	Change the disaster recovery plan manual as necessary.				

Areas to be tested

No	Activity	Status Y	Status N	Status N/A	Notes
1	Recovery of individual application systems by using files and documentation stored off-site.				
2	Reloading of system tapes and performing an IPL by using files and documentation stored off-site.				
3	Ability to process on a different computer.				
4	Ability of management to determine priority of systems with limited processing.				
5	Ability to recover and process successfully without key people.				
6	Ability of the plan to clarify areas of responsibility and the chain of command.				

K.V.P Consulting

Multi Disciplines Surrounding (MDS)

Facility Management File

שלד לבניית תיק שטח

נתונים כלליים של המתקן המאובטח :

- ☒ מיקום- כתובת מדוייקת.
- ☒ צירי הגעה למתקן.
- ☒ זהות המתקן - מהות פעילותו.
- ☒ סוג המבנה - צמוד קרקע, חלק מבית דירות, מבנה תעשייתי, שכנים דיירים וכו' .
- ☒ סביבת המבנה - סביבה עירונית/ כפרי, רח' ראשי/ צדדי, חד/ דו סטרי, אזורים ציבוריים בקרבת המבנה מרכזי קניות, תחנות אוטובוס, המצאות משרדים בבניין והותם.
- ☒ פרטי גופים מאובטחים שכנים (כולל שיטת האבטחה הנהוגה בהם).
- ☒ רשימת מספרי ודגמי כלי רכב החונים דרך קבע בקרבת המתקן (שכנים, נותני שרותים, וכו').
- ☒ שעות פעילות.
- ☒ חתך אוכלוסיית העובדים.
- ☒ חתך אוכלוסיית המבקרים/ אורחים.
- ☒ כניסות דרכי גישה רגלי/ רכוב, מעברים הכרחיים וחניונים.
- ☒ אמצעי תחבורה עמם ניתן להגיע למתקן (כולל תחבורה ציבורית).
- ☒ רכב חרום וצירי פינוי.
- ☒ מספר קומות.
- ☒ גרמי מדרגות.
- ☒ חלוקת המבנה - חדרים, חצר, מרפסת.
- ☒ סוגי הקירות במבנה - גבס, בלוקים, בטון.
- ☒ פתחי אוורור, פירים ופתחי מילוט.
- ☒ מעלית.
- ☒ נקודות כיבוי אש.
- ☒ מיקום ערכות ע"ר.
- ☒ מערכות חשמל כולל מפסקים ראשיים.
- ☒ מיקום בית חולים, תחנת משטרה, מתקנים ביטחוניים וצירי הגעה.
- ☒ נקודות תורפה בקרבת המתקן.
- ☒ מיקום הצבת חפ"ק.
- ☒ מקומות שיכולים לשמש לאיסוף מל"מ וביצוע פיגוע על המתקן - אטרקציות ציבוריים, בתי קפה, תחנות אוטובוס, בתים בבנייה, גגים, שטחים פתוחים השולטים על המתקן.
- ☒ גורמים חשושים המצויים בקרבת המתקן - זקיף, משטרה, צה"ל, מתקן מאובטח, אנשי אבטחה אזרחיים - ואמצעי זיהוי.
- ☒ הנחיות ביטחון ייחודיים למקום במקרי חירום (בצפון - ירי קטיושות, בדרום - ירי פצמרים, בשטחים - חדירת מחבלים וכו').

מפות וצילומים :

- ☒ מפה של גזרת המתקן הכוללת סימון המתקן, נקודות ציון חשובות, מעברים הכרחיים, בתי חולים.
- ☒ צילום המתקן מכיוונים שונים.
- ☒ צילום הכניסות למתקן.
- ☒ צילום כניסות לחניונים.
- ☒ צילום נקודות התורפה.
- ☒ צילום נקודות העצירה של כלי הרכב הכניסה/ יציאה מהמתקן.
- ☒ צילום נקודות/ אזורים נוספים הראויים להדגשה.

נתוני חליפת מיגון :

- ☒ סוגי דלתות, כיווני פתיחה, מיקום מפתחות.
- ☒ מיקום וסוג סורגים.
- ☒ טמ"ס - מיקום מצלמות ומוניטורים.
- ☒ לחצני מצוקה ואזעקות - מיקום ונקודות הפעלה/ נטרול, מיקום הגלאים, המוקדים אליהם האזעקה מחוברת, מספרי טלפון לתמיכה טכנית.
- ☒ מערכות בקרת כניסה טכנולוגיים - כרטיס מגנטי, קוד כניסה ורשימת מאושרי כניסה.
- ☒ מערכת כריזה.
- ☒ אדניות מיגון.
- ☒ שערים - סוג ואופן הפעלה.
- ☒ גדרות - סוג וגובה הגדר.
- ☒ תאורה - בשגרה/ בחירום.
- ☒ זכוכיות ממוגנות ירי - מיקומן.
- ☒ ציוד כיבוי אש ועזרה ראשונה.
- ☒ הגדרת חדר מבטחים/ ממ"ד.
- ☒ יציאות חירום.

נספחים :

- ☒ תצלומי אוויר.
- ☒ מפת האזור.
- ☒ תוכניות המבנה.
- ☒ תצלומים פנורמיים.
- ☒ רשימת בעלי תפקידים ומספרי טלפונים למנהלים, קב"טים ורופאי חרום והצלה.
- ☒ אמצעי חבירה וזיהוי בחירום.
- ☒ כל שינוי במבנה המתקן (שיפוצים, תוספת בנייה וכו') יעודכן בתיק השטח ויופץ לגורמים הרלוונטיים.

DRP TOC

תוכן עניינים:

BCP TOC

Business Continuity Planning Components

Getting Started **Section 1**

1. Assign departmental business continuity responsibilities.
2. Department mission and business functions/processes.
3. Identification and evaluation of scenarios, risks, events and threats.

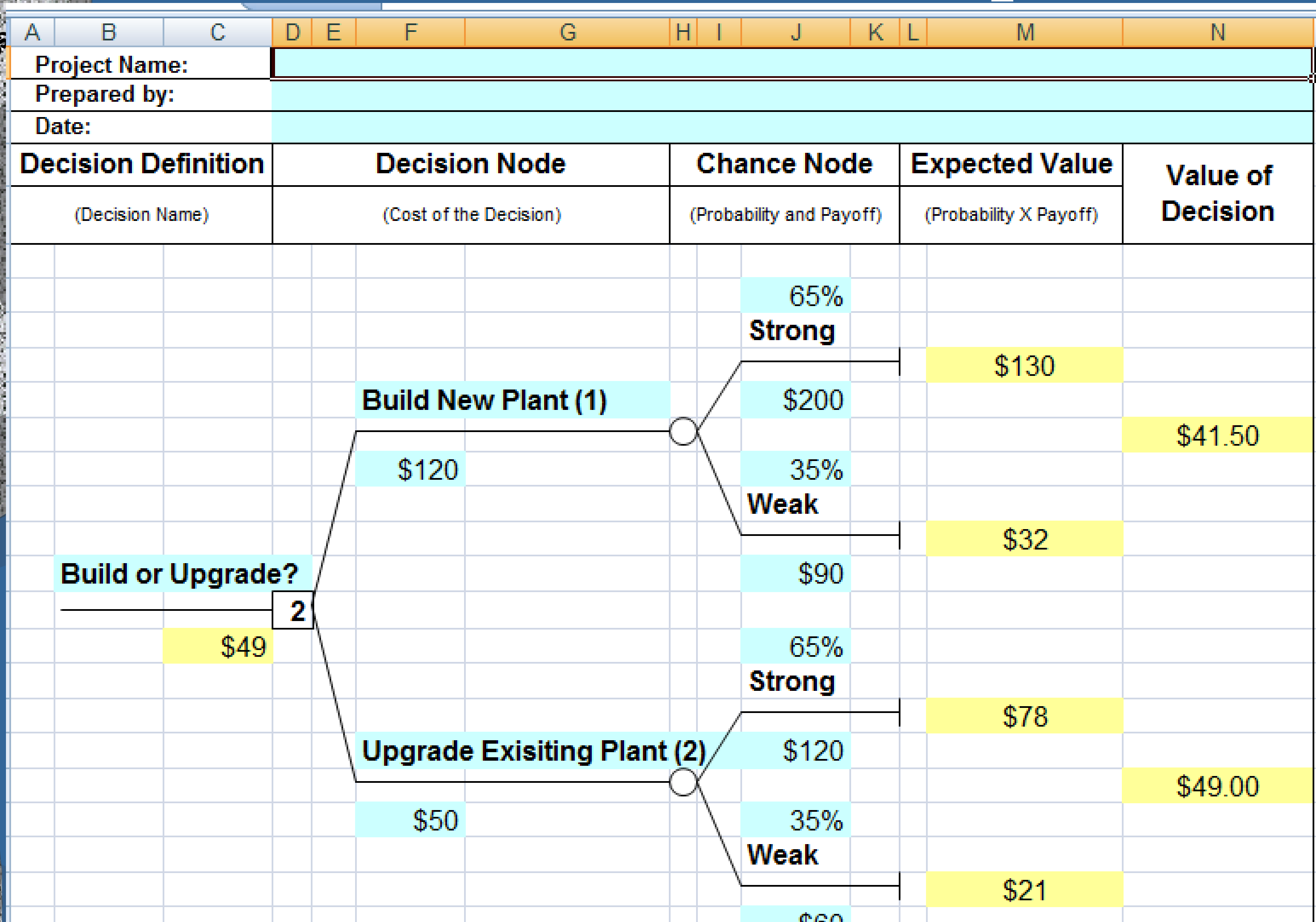
Developing the Plan **Section 2**

4. Document recovery plans to recover critical functions for each scenario.
5. Determine details to complete tasks.
6. List contact information.
7. List necessary resources and reference materials.

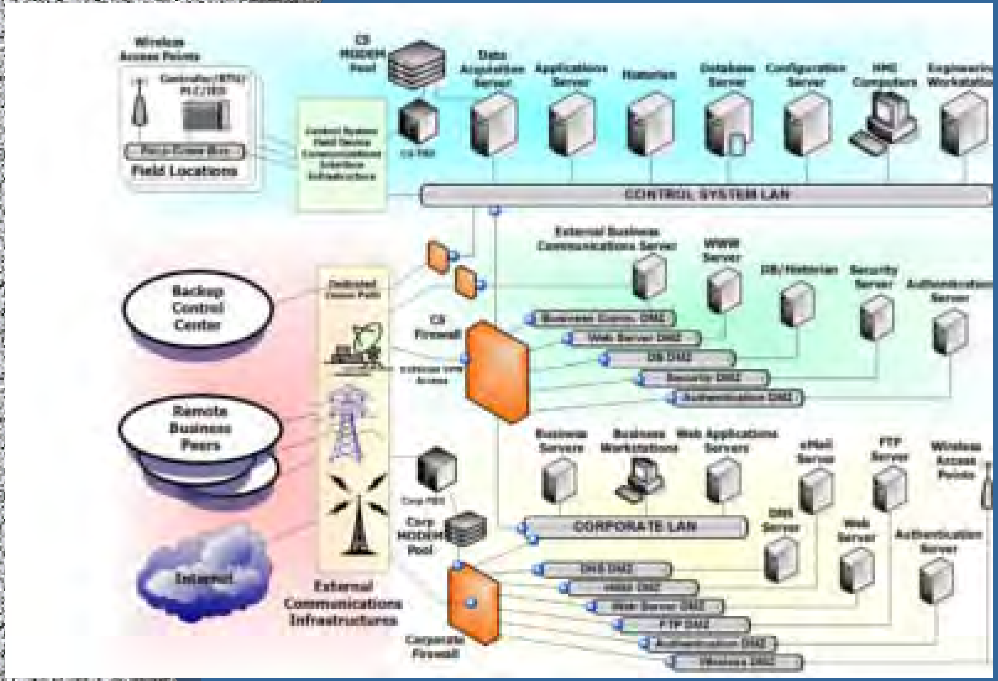
Maintaining the Plan **Section 3**

8. Train personnel on the plan.
9. Test (validate) the plan.
10. Maintain the plan.

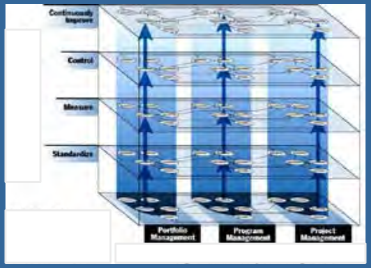
Decision Tree Template



Infrastructures and Application Mapping



Compliance Requirements to Supporting Standards Mapping



Scoping

Tool

Slides

K.V.P Consulting

CMMI – Suite
DEV
ACQ
SVC

Multi Disciplines Surrounding (MDS)

Strategy, Management, and Regulatory

Vision, planning, decision making, strategy execution,discipline, regulatory, and investment

Black text = Requirements for this level
Blue text = Descriptive characteristics or desired traits

CMMIs Interpretation

ML2

PP	PMC	M&A	PPQA	REQM	SAM	SD	AM	ARD	SSAD	DAR	OPD	OPF	IPM	OT

SGMM Levels

1	Exploring & Initiating	Developing first Smart Grid vision
		Support for experimentation
		Informal discussion with regulators
		Funding likely out of existing budget
2	Functional Investing	Integrated vision and acknowledgement
		Initial strategy and business plan approved
		Initial alignment of investments to vision
		Distinct Smart Grid Funding and budget created in collaboration with regulators and stakeholders
		Commitment to proof of concepts
		Identify initial Smart Grid leader
3	Integrating Cross Functi	Completed Smart Grid strategy and business case incorporated into Corporate strategy
		Smart Grid governance model deployed
		Smart Grid leader(s) (with authority) ensure cross-LOB application
		Mandate/consensus with regulators to make and fund Smart Grid investments
		Corporate strategy expanded
4	Optimizing Enterprise W	Smart Grid is a core competency that drives strategy and influences Corporate direction
		External stakeholders share in strategy
		Willing to invest and divest, or engage in JV and IP sharing to execute strategy

Strategy, Management | Organization, Structure | Technology | Societal & Environmental | Grid Operations | Work & Asset Management | Customer Manageme

K.V.P Consulting

Multi Disciplines Surrounding (MDS)

CMMI-SVC V 1.2

Con	Mat	Process	Process	Proc	SG N	SP N	Title
SVC	5	Process Man	Causal Anal	CAR	Purpose		The purpose of Causal Analysis and Resolution (CAR) is to identify causes of defects and problems and take action to prevent them from occurring in the future
SVC	5	Process Man	Causal Anal	CAR	1		Determine Causes of Defects and Problems Root causes of defects and problems are systematically determined
SVC	5	Process Man	Causal Anal	CAR	1	1.1	Select Defects and Problems Select defects and problems for analysis
SVC	5	Process Man	Causal Anal	CAR	1		1. Gather relevant defect and problem data
SVC	5	Process Man	Causal Anal	CAR	1		2. Determine the defects and problems to be analyzed further
SVC	5	Process Man	Causal Anal	CAR	1	1.2	Analyze Causes Perform causal analysis of selected defects and problems and propose actions to address them
SVC	5	Process Man	Causal Anal	CAR	1	1.2	1. Conduct causal analysis with those responsible for performing the task
SVC	5	Process Man	Causal Anal	CAR	1	1.2	2. Analyze selected defects and problems to determine their root causes
SVC	5	Process Man	Causal Anal	CAR	1	1.2	3. Group selected defects and problems based on their root causes
SVC	5	Process Man	Causal Anal	CAR	1	1.2	4. Propose and document actions to be taken to prevent the future occurrence of similar defects and problems
SVC	5	Process Man	Causal Anal	CAR	2		Address Causes of Defects and Problems Root causes of defects and problems are systematically addressed to prevent their future occurrence
SVC	5	Process Man	Causal Anal	CAR	2	2.1	Implement Action Proposals Implement selected action proposals developed in causal analysis
SVC	5	Process Man	Causal Anal	CAR	2	2.1	1. Analyze action proposals and determine their priorities
SVC	5	Process Man	Causal Anal	CAR	2	2.1	2. Select action proposals to be implemented
SVC	5	Process Man	Causal Anal	CAR	2	2.1	3. Create action items for implementing the action proposals
SVC	5	Process Man	Causal Anal	CAR	2	2.1	4. Identify and remove similar defects and problems that may exist in other processes and work products
SVC	5	Process Man	Causal Anal	CAR	2	2.1	5. Identify and document improvement proposals for the organization's set of standard processes
SVC	5	Process Man	Causal Anal	CAR	2	2.2	Evaluate the Effect of Changes Evaluate the effect of changes on process performance
SVC	5	Process Man	Causal Anal	CAR	2	2.2	1. Measure the change in performance of the project's defined process or of subprocesses as appropriate
SVC	5	Process Man	Causal Anal	CAR	2	2.2	2. Measure the capability of the project's defined process or of subprocesses as appropriate
SVC	5	Process Man	Causal Anal	CAR	2	2.3	Record Data Record causal analysis and resolution data for use across the project and organization
SVC	2	Support	Configurati	CM	Purpose		The purpose of Configuration Management (CM) is to establish and maintain the integrity of work products using configuration identification, configuration control, configuration status accounting, and configuration audits

CMMI-ACQ V 1.2

Con	Mat	Proc	Proc	Proc	SG N	SP Num	Title
ACQ	5	Sup	Cau	CAR			The purpose of Causal Analysis and Resolution (CAR) is to identify causes of defects and other problems and take action to prevent them from occurring in the future
ACQ	5	Sup	Cau	CAR	1	1	Determine Causes of Defects Root causes of defects and other problems are systematically determined
ACQ	5	Sup	Cau	CAR	1	1.1	Select Defect Data for Analysis Select defects and other problems for analysis
ACQ	5	Sup	Cau	CAR	1	1.1	1. Gather relevant defect or problem data
ACQ	5	Sup	Cau	CAR	1	1.1	2. Determine the defects and other problems to be analyzed further
ACQ	5	Sup	Cau	CAR	1	1.2	Analyze Causes Perform causal analysis of selected defects and other problems and propose actions to address them
ACQ	5	Sup	Cau	CAR	1	1.2	1. Conduct causal analysis with those responsible for performing the task.
ACQ	5	Sup	Cau	CAR	1	1.2	2. Analyze selected defects and other problems to determine their root causes
ACQ	5	Sup	Cau	CAR	1	1.2	3. Group selected defects and other problems based on their root causes
ACQ	5	Sup	Cau	CAR	1	1.2	4. Propose and document actions to be taken to prevent the future occurrence of similar defects or other problems
ACQ	5	Sup	Cau	CAR	2	2	Address Causes of Defects Root causes of defects and other problems are systematically addressed to prevent their future occurrence
ACQ	5	Sup	Cau	CAR	2	2.1	Implement Action Proposals Implement selected action proposals developed in causal analysis
ACQ	5	Sup	Cau	CAR	2	2.1	1. Analyze action proposals and determine their priorities
ACQ	5	Sup	Cau	CAR	2	2.1	2. Select action proposals to be implemented
ACQ	5	Sup	Cau	CAR	2	2.1	3. Create action items for implementing the action proposals
ACQ	5	Sup	Cau	CAR	2	2.1	4. Identify and remove similar defects that may exist in other processes and work products
ACQ	5	Sup	Cau	CAR	2	2.1	5. Identify and document improvement proposals for the organization's set of standard processes
ACQ	5	Sup	Cau	CAR	2	2.2	Evaluate the Effect of Changes Evaluate the effect of changes on process performance
ACQ	5	Sup	Cau	CAR	2	2.2	1. Measure the change in performance of the project's defined process or of subprocesses, as appropriate
ACQ	5	Sup	Cau	CAR	2	2.2	2. Measure the capability of the project's defined process or of subprocesses, as appropriate
ACQ	5	Sup	Cau	CAR	2	2.3	Record Data Record causal analysis and resolution data for use across the project and organization
ACQ	2	Sup	Con	CM			The purpose of Configuration Management (CM) is to establish and maintain the integrity of work products using configuration identification, configuration control, configuration status accounting, and configuration audits

CMMI-DEV V 1.2

Con	Mat	Pro	Pro	Pro	SG	SP Nur	Title
DEV	5	Sup	Cau	CAR			The purpose of Causal Analysis and Resolution (CAR) is to identi problems and take action to prevent them from occurring in the f
DEV	5	Sup	Cau	CAR	1	1	Root causes of defects and other problems are systematically det A root cause is a source of a defect such that, if it is removed, th
DEV	5	Sup	Cau	CAR	1	1.1	Select Defect Data for Analysis Select the defects and other problems for analysis
DEV	5	Sup	Cau	CAR	1	1.1	Gather relevant defect or problem data
DEV	5	Sup	Cau	CAR	1	1.1	Determine which defects and other problems will be analyzed fu
DEV	5	Sup	Cau	CAR	1	1.2	Analyze Causes Perform causal analysis of selected defects and other problems a
DEV	5	Sup	Cau	CAR	1	1.2	Conduct causal analysis with the people who are responsible for
DEV	5	Sup	Cau	CAR	1	1.2	Analyze selected defects and other problems to determine their r
DEV	5	Sup	Cau	CAR	1	1.2	Group the selected defects and other problems based on their ro
DEV	5	Sup	Cau	CAR	1	1.2	Propose and document actions that need to be taken to prevent t defects or other problems
DEV	5	Sup	Cau	CAR	2	2	Address Causes of Defects Root causes of defects and other problems are systematically add occurrence
DEV	5	Sup	Cau	CAR	2	2.1	Implement the Action Proposals Implement the selected action proposals that were developed in
DEV	5	Sup	Cau	CAR	2	2.1	1. Analyze the action proposals and determine their priorities
DEV	5	Sup	Cau	CAR	2	2.1	2. Select the action proposals that will be implemented
DEV	5	Sup	Cau	CAR	2	2.1	3. Create action items for implementing the action proposals
DEV	5	Sup	Cau	CAR	2	2.1	4. Identify and remove similar defects that may exist in other pro
DEV	5	Sup	Cau	CAR	2	2.1	5. Identify and document improvement proposals for the organi
DEV	5	Sup	Cau	CAR	2	2.2	Evaluate the Effect of Changes Evaluate the effect of changes on process performance
DEV	5	Sup	Cau	CAR	2	2.2	1. Measure the change in the performance of the project's define
DEV	5	Sup	Cau	CAR	2	2.2	2. Measure the capability of the project's defined process as app
DEV	5	Sup	Cau	CAR	2	2.3	Record Data Record causal analysis and resolution data for use across the pro
DEV	2	Sup	Con	CM			The purpose of Configuration Management (CM) is to establish a products using configuration identification, configuration contro and configuration audits

L0 Harmonization Like (16 Comm) | L1 Harmonization (Part Coom) | L2 Harmonization Other PA (Map) | CMMIs GPs | CMMI-SVC V1.2 | CMMI-ACQ V1.2 | CMMI-DEV V1.2

K.V.P Consulting

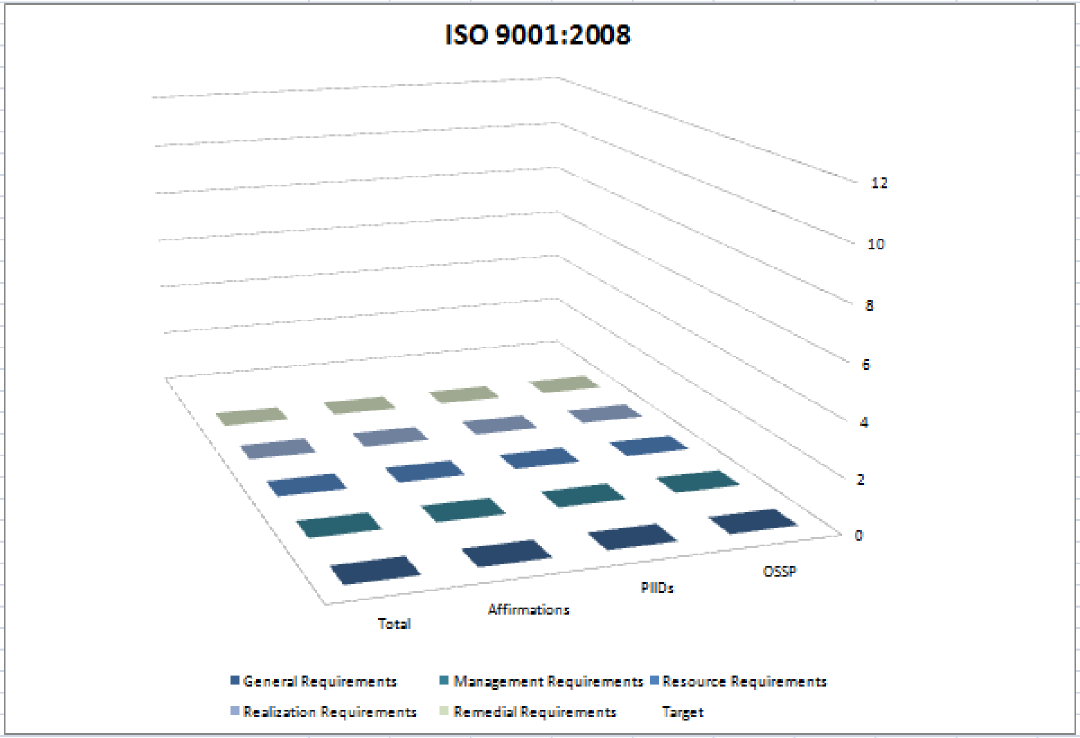
[To Index](#)

Chapter		Section			Requirements	OSSP	PIIDs	Affirmations	Total
4	General Requirements					0	0	0	0
		4.1	Develop Your Quality Management System (QMS)			0	0	0	0
				4.1.1	*Establish your organization's QMS.*	0.00	0.00	0.00	0.00
				4.1.2	*Document* your organization's QMS.	0.00	0.00	0.00	0.00
				4.1.3	*Implement* your organization's QMS.	0.00	0.00	0.00	0.00
				4.1.4	*Maintain* your organization's QMS.	0.00	0.00	0.00	0.00
				4.1.5	*Improve* your organization's QMS.	0.00	0.00	0.00	0.00
		4.2	Document Your Quality Management System (QMS)			0	0	0	0
				4.2.1	*Manage Quality Management System Documents*	*0*	*0*	*0*	*0*
				4.2.1.1	Develop documents for your organization's QMS.	0.00	0.00	0.00	0.00
				4.2.1.2	Make sure that your organization's QMS documents respect and reflect what you do and how you do it.	0.00	0.00	0.00	0.00
				4.2.2	*Prepare Quality Management System Manual*	*0*	*0*	*0*	*0*
				4.2.2.1	Establish a quality manual for your organization.	0.00	0.00	0.00	0.00
				4.2.2.2	Maintain your organization's quality manual.	0.00	0.00	0.00	0.00
				4.2.3	*Control Quality Management System Documents*	*0*	*0*	*0*	*0*
				4.2.3.1	Control your organization's QMS documents.	0.00	0.00	0.00	0.00
				4.2.3.2	Control documents that are used as QMS records.	0.00	0.00	0.00	0.00
				4.2.4	*Establish Quality Management System Records*	*0*	*0*	*0*	*0*

CMMIs SPs Kobi | CMMIs GPs | **ISO 9001 2008** | ISO 9001 Sum | OHSAS 18001 2007 | OHSAS 18001 2007 Sum | ISO9000-3 | ISO9000-3 Sum | ISO IEC 27001 2

[To Index](#)

		OSSF	PIIDs	Affirmation:	Total
		12	12	12	12
4	**General Requirements**	**0**	**0**	**0**	**0**
4.1	Develop Your Quality Management System (QMS)	0	0	0	0
4.2	Document Your Quality Management System (QMS)	0	0	0	0
5	**Management Requirements**	**0**	**0**	**0**	**0**
5.1	Show Your Commitment to Quality	0	0	0	0
5.2	Focus On Your Customers	0	0	0	0
5.3	Support Your Quality Policy	0	0	0	0
5.4	Carry Out Your QMS Planning	0	0	0	0
5.5	Allocate QMS Responsibility and Authority	0	0	0	0
5.6	Perform QMS Management Reviews	0	0	0	0
6	**Resource Requirements**	**0**	**0**	**0**	**0**
6.1	Provide Required QMS Resources	0	0	0	0
6.2	Provide Competent QMS Personnel	0	0	0	0
6.3	Provide Necessary Infrastructure	0	0	0	0
6.4	Provide Suitable Work Environment	0	0	0	0
7	**Realization Requirements**	**0**	**0**	**0**	**0**
7.1	Control Product Realization Planning	0	0	0	0
7.2	Control Customer-Related Processes	0	0	0	0
7.3	Control Product Design and Development	0	0	0	0
7.4	Control Purchasing and Purchased Products	0	0	0	0
7.5	Control Production and Service Provision	0	0	0	0
7.6	Control Monitoring and Measuring Equipment	0	0	0	0
8	**Remedial Requirements**	**0**	**0**	**0**	**0**
8.1	Establish Monitoring and Measurement Processes	0	0	0	0
8.2	Carry Out Monitoring and Measurement Activities	0	0	0	0
8.3	Identify and Control Nonconforming Products	0	0	0	0
8.4	Collect and Analyze Quality Management Data	0	0	0	0
8.5	Make Improvements and Take Remedial Actions	0	0	0	0



CMMIs SPs Kobi | CMMIs GPs | ISO 9001 2008 | **ISO 9001 Sum** | OHSAS 18001 2007 | OHSAS 18001 2007 Sum | ISO9000-3 | ISO9000-3 Sum | ISO IEC 2

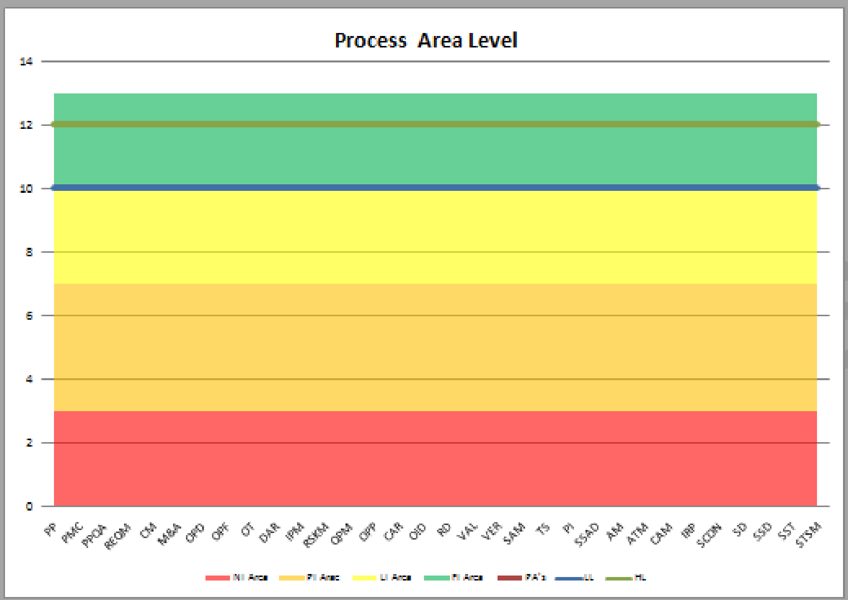
K.V.P Consulting

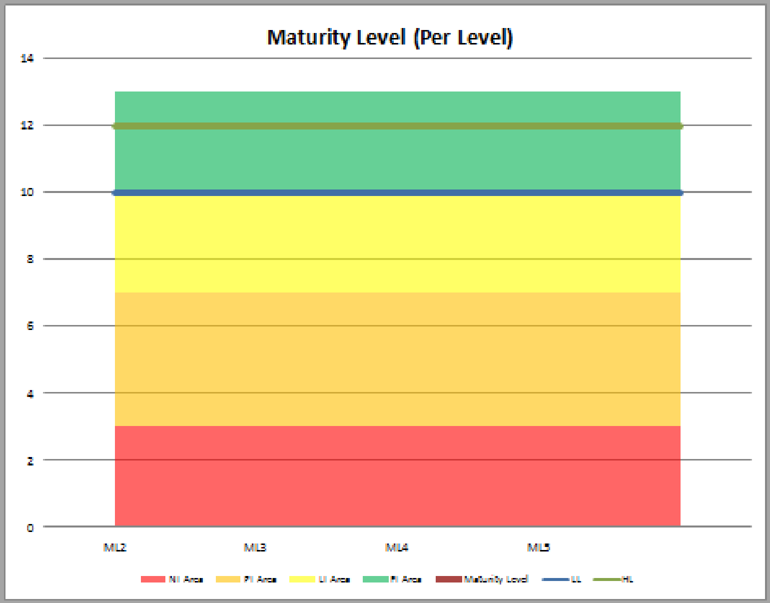
Multi Disciplines Surrounding (MDS)

Process Level	
REQM	
PP	
PMC	
M&A	
PPQA	
CM	
SAM	
RD	
TS	VAL
PI	
VER	
VAL	
OPF	
OPD	
OT	
IPM	
RSKM	
DAR	
OPP	
OPM	
OID	
CA&R	

Color	EV	Rating	LL	HL
red	1~3	NI	1.00	3.00
orange	4~6	PI	4.00	6.00
yellow	7~9	LI	7.00	9.00
green	10~12	FI	10.00	12.00

Maturity Level (Per Level)	
ML2	
ML3	
ML4	0.00
ML5	0.00





Definitions | Rating Guidelines | OSSP-ML_dashboard | OSSP-SP_dashboard | OSSP-CL_dashboard | OSSP-MLvsCL_dashboard | OSSP | CMMIs SPs Kobi | CMMIs GPs

Status

Next Steps

Pilot Results

- Verbal presentation of selected pilots

Questions